Vasco Brattka
Peter Hertling *Editors*

Handbook of Computability and Complexity in Analysis

Springer

Theory and Applications of Computability

In cooperation with the association Computability in Europe

Books published in this series will be of interest to the research community and graduate students, with a unique focus on issues of computability. The perspective of the series is multidisciplinary, recapturing the spirit of Turing by linking theoretical and real-world concerns from computer science, mathematics, biology, physics, and the philosophy of science.

The series includes research monographs, advanced and graduate texts, and books that offer an original and informative view of computability and computational paradigms.

More information about this series at http://www.springer.com/series/8819

Vasco Brattka • Peter Hertling
Editors

Handbook of Computability and Complexity in Analysis

Springer

Editors
Vasco Brattka
Deptartment of Mathematics
and Applied Mathematics
University of Cape Town
Rondebosch, South Africa

Fakultät für Informatik
Universität der Bundeswehr München
Neubiberg, Germany

Peter Hertling
Fakultät für Informatik
Universität der Bundeswehr München
Neubiberg, Germany

ISSN 2190-619X ISSN 2190-6203 (electronic)
Theory and Applications of Computability
ISBN 978-3-030-59233-2 ISBN 978-3-030-59234-9 (eBook)
https://doi.org/10.1007/978-3-030-59234-9

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

We would like to dedicate this handbook to
Klaus Weihrauch

Preface

Computable analysis is the theory of computability and complexity in analysis that arose out of Turing's seminal work [22, 23] in the 1930s. Turing's work was motivated by questions such as: which real numbers and real number functions are computable, which mathematical tasks in analysis can be solved by algorithmic means?

Nowadays this theory has many different facets that embrace topics from computability theory, algorithmic randomness, computational complexity, dynamical systems, fractals, and analog computers to logic, descriptive set theory, constructivism, and reverse mathematics. Over the previous decades computable analysis has invaded many branches of analysis, and computability and complexity questions arising from real and complex analysis, functional analysis, the theory of differential equations, (geometric) measure theory, and topology have been studied.

This handbook contains eleven chapters on computability and complexity in analysis. They cover many important research topics in computable analysis of the past two decades. Researchers and graduate students in the areas of theoretical computer science and mathematical logic will find systematic introductions to many branches of computable analysis and a wealth of information and references that will help them to navigate through the modern research literature in this field.

We thank all of the authors of the chapters in this book for their contributions. Each chapter has been reviewed according to the usual standards of our discipline. We thank the external referees for their careful work.

Readers who wish to know more about computable analysis can find a lot of information about it on the "Computability & Complexity in Analysis Network" (short: CCA Network): http://cca-net.de/, for example a list of relevant conferences and workshops, a bibliography, and a list of the members of the network.

Finally, we would like to dedicate this handbook to Klaus Weihrauch. His efforts played a central role in the tremendous growth of the field since the 1990s.

A Very Short History of Computable Analysis in the Previous Century

In this section we give a very short overview of the development of computable analysis in the twentieth century. We restrain ourselves from citing any of the many important original research papers in this area with two exceptions: the seminal papers [22, 23] by Turing. For a more detailed historical account including many references the reader is referred to [2].

Computation problems have been solved for thousands of years. In fact, the desire to solve computation problems that arose in everyday life and in other scientific disciplines has been the main driving force behind the development of mathematics throughout its history. But it was not before the twentieth century that the process of solving computation problems itself was analyzed in detail. The first person who did this was Alan Turing in his seminal paper "On Computable Numbers, with an Application to the Entscheidungsproblem" [22]. In this paper he developed the notion of a *Turing machine*, which is still considered the most important theoretical model of a digital computer. The Turing machine model is the basis of both computability theory and structural complexity theory. It offers mathematically precise notions of what it means to solve a computation problem and what it means to solve a computation problem within a certain amount of time, e.g., within polynomial time, or with a certain amount of workspace. Interestingly, in both fields just mentioned, computability theory (which has been called "recursion theory" for a long time) and structural complexity theory, most of the computation problems that were analyzed using the notions offered by the Turing machine model were problems over the natural numbers, in recursion theory in particular questions involving sets of natural numbers, or over other discrete structures. Problems involving objects like real numbers or real functions, so problems from (numerical) analysis or from other parts of the "continuous world" of mathematics, were studied as well, but to a much smaller extent. This is surprising as the paper [22] itself was explicitly concerned with the question of which real numbers should be called computable. This question was, as it seems, the main motivation for Turing to develop his machine model. In this paper and in a correction [23] that appeared shortly afterwards Turing gave definitions of computable real numbers and computable real functions and made first observations about these objects. So, one may call these papers also the first papers concerned with computability in analysis. Of course, there were already other scientists before Turing who worked on ideas concerning computability of real numbers or real functions. One should mention Émil Borel, Luitzen E. J. Brouwer, and work concerned with primitive recursion in the 1920s. But the precise mathematical notions concerning computability that are needed for a satisfying theory of computability in analysis were first provided by Turing.

Turing's ideas concerning the computability of real numbers and functions were taken up by Banach and Mazur already before the Second World War. Some notes by Mazur on this were published only much later [16], though. Banach and Mazur called a function mapping real numbers to real numbers computable if it maps any

computable sequence of real numbers to a computable sequence of real numbers. This idea was taken up again later. After the Second World War several people across the world worked on computability questions concerning real numbers or real functions. For example Ernst Specker in Switzerland constructed the so-called *Specker sequence*, a strictly increasing computable sequence of rational numbers converging to a noncomputable real number. Already in the papers [22, 23] by Turing different representations of real numbers had been considered. The questions concerning different representations of real numbers were taken up by Specker and other people, for example by Andrzej Mostowski. In the 1950s Georg Kreisel and H. Gordon Rice obtained some first results in computable analysis as well. In a series of papers Andrzej Grzegorczyk laid the foundations for the "Polish school of computable analysis". In particular, he gave several characterizations of a computability notion for real functions that may be called *effective continuity*. These notions were the basis for several later developments in computable analysis. At the same time Daniel Lacombe in France introduced effectivity notions for open sets of real numbers and was the first to extend computability notions from Euclidean spaces to other, more general topological spaces. Also at the same time, Andrej A. Markov founded the "Russian school of constructive mathematics", with contributions by Andrej A. Markov, Nikolai Šanin, Igor D. Zaslavskiĭ, Gregory Ceĭtin, Vladimir P. Orevkov, Osvald Demuth, Boris Kushner, and many others. This school combined ideas from constructive mathematics with notions from computability theory. If one ignores the constructive aspects, one may say that in this theory a function mapping computable real numbers to computable real numbers is called computable if there is an effective procedure for mapping a program for the input real number to a program for the output real number. For an overview the reader is referred to Šanin's book [20] or Kushner's book [15]. Oliver Aberth's work [1] in the 1960s and 1970s is based on similar ideas and notions, though developed not in a constructive framework but in a classical computability-theoretic framework. In the book [3] Errett Bishop developed a considerable part of analysis in a constructive way. A revised and expanded version [4] appeared in 1985. Although these books are neither based on computability-theoretic notions nor intended to be a contribution to computable analysis, the constructive proofs in these books can easily be turned into computability-theoretic statements. The influential approach to computable analysis by Marian Boykan Pour-El and by J. Ian Richards was based on one of the characterizations of computable functions by Grzegorczyk and culminated in their book [19]. They were in particular concerned with computability on Banach spaces and with the solvability of differential equations in the strict sense of computability theory. In the 1980s Harvey Friedman and Ker-I Ko started to work on complexity-theoretic aspects of real number computations. Their results are summarized in another influential book [12], by Ker-I Ko. Slightly earlier Hauck in East Germany had taken up the topic of representations of real numbers and extended it to representations of other objects. This idea was taken up by Klaus Weihrauch and his Ph.D. student Christoph Kreitz who developed a systematic theory of representations of certain topological spaces and showed how to develop computability notions for general spaces using *admissible* representations and computability-theoretic notions. For an

overview of computable analysis from this point of view the reader is referred to the monograph [24] by Weihrauch. Shorter reviews of computable analysis are [6] and [5]. Since then computable analysis has seen a rather rapid and broad development. It is the purpose of this handbook to give the reader an overview of many important current research topics in the general area of computability and complexity in analysis.

Overview of the Chapters

We have grouped the eleven chapters under the following headings:

1. Computability in Analysis
2. Complexity, Dynamics, and Randomness
3. Constructivity, Logic, and Descriptive Complexity

Computability in Analysis

The four chapters in the part about "Computability in Analysis" deal with computability questions concerning objects in classical areas of analysis: real numbers, subsets of metric spaces, differential equations, and complex analysis.

One of the fundamental notions of computable analysis is the notion of a computable real number. We already mentioned that Specker had shown that there exist noncomputable real numbers that can nevertheless be approximated effectively from one side, the so-called *left-computable* or *c.e.* real numbers. In the chapter "Computability of Real Numbers" Robert Rettinger and Xizhong Zheng review these and many other classes of real numbers that can be approximated in some effective way and that have been analyzed during the past decades.

We also mentioned that Lacombe had started to analyze computability conditions for open subsets of Euclidean space. In the chapter "Computability of Subsets of Metric Spaces" Zvonko Iljazović and Takayuki Kihara give a survey of results related to computability conditions for subsets of metric spaces. They discuss various effective topological properties, computability of points in co-c.e. closed sets, classifications of Polish spaces, the question of when semicomputable sets are actually computable, computability structures, and many other topics.

In the chapter "Computability of Differential Equations" Daniel Graça and Ning Zhong give a survey of computability-theoretic and complexity-theoretic results concerning differential equations. They study when the solution of an initial value problem for an ordinary differential equation is computable and present results concerning the computational complexity of a computable solution. They also study the asymptotic behavior of solutions of ordinary differential equations and several important partial differential equations.

Valentin V. Andreev and Timothy H. McNicholl present an introduction to the field of "Computable Complex Analysis". They discuss computability-theoretic and complexity-theoretic results on conformal mappings, harmonic functions, infinite products, and constants, and they present several open problems.

Complexity, Dynamics, and Randomness

The first chapter in this part of the book, the chapter "Computable Geometric Complex Analysis and Complex Dynamics" by Cristóbal Rojas and Michael Yampolsky, deals with two special parts of computable complex analysis that have been the subject of a lot of activity during the past two decades: on the one hand, the computability and computational complexity of conformal mappings and their boundary extensions and, on the other hand, the computability and computational complexity of objects in complex dynamics, in particular of Julia sets. The article complements the book [7] by Braverman and Yampolsky on the computability on Julia sets.

A different topic and, in fact, different computation models, are discussed in the chapter "A Survey on Analog Models of Computation" by Olivier Bournez and Amaury Pouly. They discuss many different analog models of computation, mostly from the point of view of computation theory.

The fundamental notion of an algorithmically random or *Martin-Löf random* binary sequence was invented in the 1960s by Martin-Löf. One should also mention Solomonoff, Kolmogorov, and Chaitin in this context. In the 1990s and the first decade after 2000 the research area of algorithmic randomness was extremely active, leading to two textbooks [17, 10]. A recent survey article highlighting relations between this area and computable analysis is [18]. The chapter "Computable Measure Theory and Algorithmic Randomness" by Mathieu Hoyrup and Jason Rute gives a compact overview of recent results in computable measure and probability theory and in algorithmic randomness.

In the chapter "Algorithmic Fractal Dimensions in Geometric Measure Theory" Jack H. Lutz and Elvira Mayordomo survey the development of the theory of algorithmic fractal dimensions in the past two decades. They discuss the interactions of this theory with geometric measure theory, in particular with fractal geometry, and its connections with computable functions on the reals. And they present recent results that show how one can use algorithmic dimensions for proving new theorems in classical fractal geometry.

Constructivity, Logic, and Descriptive Complexity

We already mentioned that representations of real numbers have played an important role in computable analysis from its very beginnings, namely already in the papers [22, 23] by Turing. In the chapter "Admissibly Represented Spaces and Qcb-

Spaces" Matthias Schröder presents an overview of the state of the art concerning representations of topological spaces, with an emphasis on admissibly represented spaces and the closely related class of qcb-spaces.

We also mentioned that there are various versions of constructive analysis. Two quite recent books on constructive analysis are [8, 9]. In the chapter "Bishop-Style Constructive Reverse Mathematics" Hannes Diener and Hajime Ishihara present an overview of the current state of the area called constructive reverse mathematics. It is the goal of this area to put some order into principles and results from classical mathematics that cannot be proved in Bishop-style constructive mathematics. This area is related to Simpson-style reverse mathematics [21] and to the topic of the final chapter of the handbook.

The final chapter of the handbook, "Weihrauch Complexity in Computable Analysis" by Vasco Brattka, Guido Gherardi, and Arno Pauly, is a self-contained introduction to the currently very active area of Weihrauch complexity and its applications in computable analysis. The goal of this theory is to order certain mathematical theorems according to their computational strength. The tool used for this is Weihrauch reducibility. The chapter also contains a discussion of the relation of this approach to other approaches.

Outlook

There are more topics that would have deserved to be treated in chapters in this book. For example, the area of "proof mining" is related to computable analysis. In this area classical proofs of theorems are analyzed with the goal to obtain effective bounds or effective rates of convergence. The interested reader can find more about this in the survey article [14] and in the monograph [13]. Also, connections to other versions of constructive mathematics, to realizability, and to domain theory could not be included. The perhaps most regrettable gap is that we were not able to include a chapter on the recent developments in uniform computational complexity in analysis (see for instance [11]). Finally, the fundamental motivation for computable analysis is the hope that it leads to improvements in the practice of computing. But the practice of computing is a vast area spreading in many different directions, e.g., the development and implementation of numerical algorithms, error analysis, interval analysis, high precision arithmetic, programming languages, software libraries, computer arithmetic, and many others. Therefore, we have decided not to include chapters reaching out to these topics.

We hope that the interested reader will find this book useful for getting an overview of some of the currently most important research directions and hottest research topics in computable analysis. Enjoy!

Munich, *Vasco Brattka*
December 2020 *Peter Hertling*

References

1. Aberth, O.: Computable Analysis. McGraw-Hill, New York (1980)
2. Avigad, J., Brattka, V.: Computability and analysis: the legacy of Alan Turing. In: R. Downey (ed.) Turing's Legacy: Developments from Turing's Ideas in Logic, *Lecture Notes in Logic*, vol. 42, pp. 1–47. Cambridge University Press, Cambridge, UK (2014)
3. Bishop, E.: Foundations of Constructive Analysis. McGraw-Hill, New York (1967)
4. Bishop, E., Bridges, D.S.: Constructive Analysis, *Grundlehren der Mathematischen Wissenschaften*, vol. 279. Springer, Berlin (1985)
5. Brattka, V., Hertling, P., Weihrauch, K.: A tutorial on computable analysis. In: S.B. Cooper, B. Löwe, A. Sorbi (eds.) New Computational Paradigms: Changing Conceptions of What Is Computable, pp. 425–491. Springer, New York (2008)
6. Braverman, M., Cook, S.A.: Computing over the reals: Foundations for scientific computing. Notices of the AMS **53**(3), 318–329 (2006)
7. Braverman, M., Yampolsky, M.: Computability of Julia Sets, *Algorithms and Computation in Mathematics*, vol. 23. Springer, Berlin (2008)
8. Bridges, D.S., Vîţă, L.S.: Techniques of Constructive Analysis. Springer, New York (2006)
9. Bridges, D.S., Vîţă, L.S.: Apartness and Uniformity. *Theory and Applications of Computability*. Springer, Heidelberg (2011)
10. Downey, R.G., Hirschfeldt, D.R.: Algorithmic Randomness and Complexity. *Theory and Applications of Computability*. Springer, New York (2010)
11. Kawamura, A., Cook, S.A.: Complexity theory for operators in analysis. ACM Transactions on Computation Theory **4**(2), 5:1–5:24 (2012)
12. Ko, K.I.: Complexity Theory of Real Functions. *Progress in Theoretical Computer Science*. Birkhäuser, Boston (1991)
13. Kohlenbach, U.: Applied Proof Theory: Proof Interpretations and Their Use in Mathematics. Springer, Berlin (2008)
14. Kohlenbach, U.: Effective bounds from proofs in abstract functional analysis. In: S.B. Cooper, B. Löwe, A. Sorbi (eds.) New Computational Paradigms: Changing Conceptions of What Is Computable, pp. 223–258. Springer, New York (2008)
15. Kushner, B.A.: Lectures on Constructive Mathematical Analysis, *Translations of Mathematical Monographs*, vol. 60. American Mathematical Society, Providence (1984)
16. Mazur, S.: Computable Analysis, *Razprawy Matematyczne*, vol. 33. Pánstwowe Wydawnictwo Naukowe, Warsaw (1963)
17. Nies, A.: Computability and Randomness, *Oxford Logic Guides*, vol. 51. Oxford University Press, New York (2009)
18. Nies, A.: Lowness, randomness, and computable analysis. In: A. Day, M. Fellows, N. Greenberg, B. Khoussainov, A. Melnikov, F. Rosamond (eds.) Computability and Complexity: Essays Dedicated to Rodney G. Downey on the Occasion of His 60th Birthday, *Lecture Notes in Computer Science*, vol. 10010, pp. 738–754. Springer, Cham (2017)
19. Pour-El, M.B., Richards, J.I.: Computability in Analysis and Physics. *Perspectives in Mathematical Logic*. Springer, Berlin (1989)
20. Šanin, N.: Constructive Real Numbers and Constructive Function Spaces, *Translations of Mathematical Monographs*, vol. 21. American Mathematical Society, Providence (1968)
21. Simpson, S.G.: Subsystems of Second Order Arithmetic, 2nd edn. *Perspectives in Logic*. Cambridge University Press (2009)
22. Turing, A.M.: On computable numbers, with an application to the "Entscheidungsproblem". Proceedings of the London Mathematical Society **42**(2), 230–265 (1937)
23. Turing, A.M.: On computable numbers, with an application to the "Entscheidungsproblem". A correction. Proceedings of the London Mathematical Society **43**(2), 544–546 (1938)
24. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)

Contents

List of Contributors

Valentin V. Andreev
Department of Mathematics, Lamar University, Beaumont, Texas, USA
e-mail: valentin.andreev@lamar.edu

Olivier Bournez
École Polytechnique, Palaiseau, France
e-mail: bournez@lix.polytechnique.fr

Vasco Brattka
Faculty of Computer Science, Universität der Bundeswehr München, Germany and Department of Mathematics & Applied Mathematics, University of Cape Town, South Africa
e-mail: Vasco.Brattka@cca-net.de

Hannes Diener
School of Mathematics and Statistics, University of Canterbury, Christchurch, New Zealand
e-mail: hannes.diener@canterbury.ac.nz

Guido Gherardi
Dipartimento di Filosofia e Comunicazione, Università di Bologna, Italy
e-mail: Guido.Gherardi@unibo.it

Daniel S. Graça
Universidade do Algarve, Faro, Portugal and Instituto de Telecomunicações, Portugal
e-mail: dgraca@ualg.pt

Mathieu Hoyrup
Université de Lorraine, CNRS, Inria, LORIA, Nancy, France
e-mail: mathieu.hoyrup@inria.fr

Zvonko Iljazović
Department of Mathematics, Faculty of Science, University of Zagreb, Croatia
e-mail: zilj@math.hr

Hajime Ishihara
School of Information Science, Japan Advanced Institute of Science and Technology, Nomi, Japan
e-mail: ishihara@jaist.ac.jp

Takayuki Kihara
Department of Mathematical Informatics, Graduate School of Informatics, Nagoya University, Japan
e-mail: kihara@i.nagoya-u.ac.jp

Jack H. Lutz
Department of Computer Science, Iowa State University, Ames, Iowa, USA
e-mail: lutz@cs.iastate.edu

Elvira Mayordomo
Departamento de Informática e Ingeniería de Sistemas, Instituto de Investigación en Ingeniería de Aragón, Universidad de Zaragoza, Spain
e-mail: elvira@unizar.es

Timothy H. McNicholl
Department of Mathematics, Iowa State University, Ames, Iowa, USA
e-mail: mcnichol@iastate.edu

Arno Pauly
Department of Computer Science, Swansea University, United Kingdom
e-mail: a.m.pauly@swansea.ac.uk

Amaury Pouly
Max Planck Institute for Software Systems, Saarbrücken, Germany
e-mail: pamaury@mpi-sws.org

Robert Rettinger
Department of Computer Science, Fachhochschule Dortmund, Germany
e-mail: robert.rettinger@fh-dortmund.de

Cristóbal Rojas
Departamento de Matemáticas, Universidad Andrés Bello, Santiago, Chile
e-mail: crojas@mat-unab.cl

Jason Rute
Department of Mathematics, Pennsylvania State University, State College, Pennsylvania, USA
e-mail: jason.rute@uwalumni.com

Matthias Schröder
Department of Mathematics, TU Darmstadt, Germany
e-mail: matthias.schroeder@cca-net.de

Michael Yampolsky
Department of Mathematics, University of Toronto, Canada
e-mail: yampol@math.toronto.edu

Xizhong Zheng
Department of Computer Science and Mathematics, Arcadia University, Glenside, Pennsylvania, USA
e-mail: zhengx@arcadia.edu

Ning Zhong
Department of Mathematical Sciences, University of Cincinnati, Ohio, USA
e-mail: ning.zhong@uc.edu

Part I
Computability in Analysis

Chapter 1
Computability of Real Numbers

Robert Rettinger and Xizhong Zheng

Abstract In scientific computation and engineering real numbers are typically approximated by rational numbers which approximate, in principle, the real numbers up to any given precision. This means that they are treated as the limits of computable sequences of rational numbers where an effective error estimation is often expected. Although such approximations exist for many constants, they do not exist for all constants. More precisely, approximations with an effective control over the approximation error exist only for the *computable* real numbers. As long as we are interested in approximations of real numbers, the weakest condition we can ask for is that a real number can be approximated by a computable sequence of rational numbers without any information about the approximation error. These real numbers are called *computably approximable*. By relaxing and varying conditions on the knowledge one has about the approximation, one gets several natural classes between these two classes of real numbers. In this paper we review the most natural classes which have been investigated over the past decades with respect to this viewpoint. Most of these classes can be characterized in different ways, partly by purely mathematical properties and partly by their computational properties.

1.1 Introduction

A rigorous theory of real numbers is regarded as a solid foundation of modern mathematics as well as modern science in general. Typically real-world problems are

Robert Rettinger
Department of Computer Science, Fachhochschule Dortmund, Germany
e-mail: robert.rettinger@fh-dortmund.de

Xizhong Zheng
Department of Computer Science and Mathematics, Arcadia University, Glenside, Pennsylvania, USA
e-mail: zhengx@arcadia.edu

V. Brattka, P. Hertling (eds.), *Handbook of Computability and Complexity in Analysis*,
Theory and Applications of Computability, https://doi.org/10.1007/978-3-030-59234-9_1

modeled and solved by continuous structures which involve real numbers in some way. Usually, various computations related to real numbers are involved. Thanks to the rapid development of modern computer technology, most of these computations of or on real numbers can be done to a very satisfying precision. Today's computers are so powerful that users can hardly recognize the fact that we can compute only by means of approximations when we actually want to compute on real numbers. Nevertheless, a gap remains between the well-established mathematical theory of real numbers and the related continuous structures on one side, and the discrete essence of computation by means of modern computers on the other side. To bridge this gap, a theory of computability of and on real numbers is of theoretical as well as practical importance.

Thus it is not surprising that already in the early ages of computer science the question of a reasonable definition of the computable real numbers was raised: such a definition was the motivation of Alan Turing's seminal paper [41] in which the Turing machine model was introduced. According to Turing [41], "a (real) number is computable if its decimal can be written down by a machine", in other words, a real number is computable if its decimal digits can be represented by a computable function. After Turing's definition, Specker [40] compared the definitions of "primitive recursive real numbers" (he called them "*recursive real numbers*") and the "computable real numbers" in terms of computable Cauchy sequences. It turns out that four definitions of primitive recursive real numbers based on Cauchy sequences, decimal expansions, Dedekind cuts and nested interval sequences, respectively, are all different! In particular, the Dedekind primitive recursive real numbers are not closed under addition, and the decimal primitive recursive real numbers are not closed under multiplication.

For the computable real numbers, the situation is completely different. As was pointed out by Robinson [36], Myhill [23], Rice [35] and others, all classical representations of real numbers including Cauchy sequences, Dedekind cuts, etc. lead to the same computability notion of real numbers as Turing's. However, if we want to define computable real functions, only the approach based on the representation by Cauchy sequences seems most suitable. In this case, a real number x is called *computable* if there is a computable sequence (x_s) of rational numbers which converges to x with an effective error estimation. In other words,

$$(\forall s)(|x - x_s| \leq 2^{-s}), \tag{1.1}$$

where a sequence (x_n) of rational numbers is called *computable* if there are three computable functions $f, g, h : \mathbb{N} \to \mathbb{N}$ such that $x_n = (f(n) - g(n))/(h(n) + 1)$ for all natural numbers n. The class of computable real numbers is denoted by **EC** (for **E**ffective **C**omputable).

The effective error estimation condition (1.1) can equivalently be replaced by the following condition

$$(\forall s)(|x_s - x_{s+1}| \leq 2^{-s}). \tag{1.2}$$

This is a pure property of the sequence without mentioning its limit explicitly. A sequence (x_s) *converges effectively* if it satisfies Condition (1.2). By relaxing this condition various versions of weakly computable real numbers can be defined accordingly. Depending on the convergence quality of a computable sequence of rational numbers, its limit will have different levels of computability. One way to measure the convergence quality is to count how often the condition (1.2) will be destroyed. More reasonably, we can count the number of non-overlapping index pairs (i, j) such that $|x_i - x_j| > 2^{-n}$ for a given n. For example, if the number of such index pairs is bounded by a computable function of the argument n, then we will have a quite natural class of real numbers of some weaker computability. In addition, we are not only interested in the individual weakly computable real numbers, but also the whole set of all real numbers of the same computability level. In particular, we will explore which mathematical properties these subsets of weakly computable real numbers can have.

In this chapter we will discuss some known classes of the real numbers with respect to this viewpoint, i.e., by means of the knowledge we have on the convergence of a given, computable sequence of rational numbers. We will furthermore give both mathematical and computational properties of the classes we present and stress the fact that indeed most of these classes have, in addition to their nice computational properties, also natural mathematical properties.

This is not the only viewpoint one can take to classify the real numbers. A different viewpoint is to classify the real numbers by means of (in)compressibility, i.e., Kolmogorov complexity. This viewpoint leads to another interesting and deep classification of the real numbers including such prominent notions as randomness and K-triviality. The classification by means of Kolmogorov complexity leads usually to a totally different one compared to the viewpoint taken in this chapter. However, both viewpoints have common points of intersection. We will introduce this theory only so far as it influences the classification by means of our viewpoint. Another viewpoint is by pure means of classical computability theory: once a representation, such as the decimal representation we are used to, is chosen, one can easily translate most results of classical computability theory to real numbers.

We will start out with a brief overview of probably the most important class: the class of computable real numbers. Afterwards we will give an overview of the central classes of real numbers defined by different restrictions on the convergence of a computable sequence of rational numbers. These classes will then be discussed in the following sections in some more detail.

1.2 Computable Real Numbers

The class of computable real numbers is probably the most important and natural one among the considered classes. It contains all constants which we can use in real computations. For most of the constants we are used to, such as the number π or Euler's constant e, we know how to compute approximations up to very high precisions

very efficiently. Another class of real numbers which we can easily compute are the algebraic numbers. However, not every constant used in mathematics is known to be computable. For Bloch's constant, for example, mathematical methods so far give only very rough bounds. In the last couple of years some of these constants were proven to be computable, whereas others such as the Hayman-Wu-constant [14] are not known to be computable at all.

To compute mathematical constants one usually requires mathematical constructions together with error estimations, where both are usually either directly given by simple constructions or equations or given by images of functions, which are, in the best case, even analytic. A quite recent result by Hertling and Spandl [15] shows that such constructions and error estimations can be quite involved. They proved the polynomial-time computability of the Feigenbaum constant, actually as a by-product. A different technique, which can be applied to several other problems as well (see [30]), was used to prove computability of Bloch's and Landau's constants [28, 29]. We do not have the space to explain the details here and thus give rather some overall intuition on the proof technique. To start with, Landau's constant λ is a fundamental constant in complex analysis. It is known that there exist constants $r > 0$ such that for any analytic function $h : \mathbb{D} \to \mathbb{C}$ from the unit disc into the complex plane there exists a disc of radius at least $|h'(0)| \cdot r$ inside the image $h(\mathbb{D})$. Landau's constant is the largest of these constants. By Robinson and Rademacher [26] it was shown that

$$0.5 \leq \lambda < 0.54325...$$

Bloch's constant is defined similarly where the image disc is a conformal image. To prove computability of these constants essentially two steps were used: First it was shown how the radius of the largest (conformal) disc inside an image $h(\mathbb{D})$ for every holomorphic function f can be approximated up to any precision. This step needs different techniques for Bloch's and Landau's constants. In the second step a compact space of holomorphic functions is defined in such a way that searching this space can be effectively done and that the approximations of the first step lead to approximations of the neighborhood. This reduces the problem to finitely many approximations solved by the first step.

Besides the fact that most interesting single real numbers belong to the class of computable real numbers, this class has a rich structure by means of restricting the resources of the computational model. We have already mentioned that this can be done by means of classical notions of recursion theory such as primitive recursive functions. Another way of restricting resources is by means of complexity theory. Classes like P and NP can be defined nearly literally once one has decided on one of the representations. Unfortunately, however, different representations can indeed lead to different notions of complexity classes. Consider the set of dyadics, i.e., numbers $z/2^k$ for integers z and k and some standard representation over $\{0,1,.\}$, e.g., by words 1001.10111 with the usual meaning. Furthermore let the left cut *LC* of a real x be the set of dyadics d with $d < x$. We say that x has a *polynomial-time computable left cut* iff *LC* is in P. We say that x has a *polynomial-time computable Cauchy sequence* iff there exists a polynomial-time computable function mapping

0^n to some dyadic d_n such that $|d_n - x| < 2^{-n}$. We say that x has a *polynomial-time decimal representation* iff the function which maps 0^n to the decimal representation of x up to length n is polynomial-time computable. The following result was shown by Ker-I Ko in [16], where $\mathbf{LC_P}$, $\mathbf{CS_P}$ and $\mathbf{DR_P}$ denote the set of real numbers with polynomial-time computable left cut, with polynomial-time computable Cauchy sequence and with polynomial-time computable decimal representation, respectively.

Theorem 1.2.1 (Ker-I Ko [16]). $\mathbf{LC_P} = \mathbf{DR_P} \subsetneq \mathbf{CS_P}$

Classical results such as dense time hierarchies as well as results on classes defined by means of time-bounded functions with certain mathematical restrictions, e.g., by roots of such functions, were also shown for real numbers (see [22]). Finally even theorems from abstract complexity theory such as the Speed-Up and Gap Theorems were adapted to the real numbers (see [18]).

1.3 A Finite Hierarchy of Computably Approximable Real Numbers

The computability of a real number requires an effective error estimation of the computable approximation by rational numbers. Several weaker conditions on the convergence of computable approximations have been investigated. The first and the most popular condition is the monotone approximation. We call a real number x *computably enumerable*, (or *c.e.*, for short), if there is an increasing computable sequence (x_s) of rational numbers which converges to x. Equivalently, a real number x is c.e. if its left Dedekind cut is a c.e. set. This is also the reason why we call such real numbers c.e. The class of c.e. real numbers is denoted by **CE**. A c.e. number can be approximated by rational numbers from the left side on the real line. Therefore, c.e. real numbers are also called *left computable*. C.e. real numbers are also widely called *left-c.e.* (see [7]).

Naturally, we can also define the *right computable* or *co-c.e.* real numbers in a corresponding way. C.e. and co-c.e. real numbers together are called *semi-computable* and we use **SC** to denote the class of all semi-computable real numbers. Semi-computable real numbers also have a unified characterization: a real number x is semi-computable iff there is a computable sequence (x_s) of rational numbers which converges to x (1-)*monotonically* in the sense that $|x - x_t| \leq |x - x_s|$ for all $t > s$ (see [1, 43] for more details).

Without loss of generality we can consider only the real numbers in the unit interval $[0,1]$ in this chapter. Thus, the binary expansion of a real number x corresponds to a set A of natural numbers in a very natural way: $x = x_A := \sum_{n \in A} 2^{-(n+1)}$. We will simply call the set A the *binary expansion* of the real number x if $x = x_A$. The c.e. real numbers are regarded as the counterparts of the c.e. sets of natural numbers in computability theory. Apparently, a real number of a c.e. binary expansion must be c.e. However the converse does not hold, as observed by Jockusch (see [37]). For example, let A be a non-computable c.e. set with a computable enumeration

$A = \{a_1, a_2, a_3, \cdots\}$. Then the set $B := A \oplus \overline{A} := \{2n : n \in A\} \cup \{2n+1 : n \notin A\}$ is not a c.e. set. On the other hand, we have

$$x_B = \sum_{i \in A} 2^{-2i-1} + \sum_{i \notin A} 2^{-2i-2} = \sum_{i \in A} 2^{-2i-1} + \left(\frac{1}{3} - \sum_{i \in A} 2^{-2i-2}\right) = \frac{1}{3} + \sum_{i \in A} 2^{-2i-2}.$$

Therefore, the computable sequence (x_s) of rational numbers defined by

$$x_s := \frac{1}{3} + \sum_{i=1}^{s} 2^{-(2a_i+2)}$$

is increasing and it converges to x_B, and hence x_B is a c.e. real number.

As a generalization of Jockusch's example [37], a precise characterization of c.e. real numbers in terms of binary expansions is shown by Calude, Hertling, Khoussainov and Wang in [4] by strongly ω-c.e. sets. According to Ershov [10, 11, 12], a set $A \subseteq \mathbb{N}$ is called *h-c.e.* for a function $h : \mathbb{N} \to \mathbb{N}$ if there is a computable sequence (A_s) of finite sets which converges to A such that $A_0 = \emptyset$ and

$$(\forall n)(|\{s \in \mathbb{N} : A_s(n) \neq A_{s+1}(n)\}| \leq h(n)). \tag{1.3}$$

That is, in the approximation (A_s) to A, any number n can be put into and deleted from A no more than $h(n)$ times. A set is called *ω-c.e.* if it is h-c.e. for a computable function h. As a variation, a set A is called *strongly ω-c.e.* if there is a computable sequence (A_s) of finite sets which converges to A such that

$$(\forall n)(\forall s)(n \in A_s \backslash A_{s+1} \Longrightarrow (\exists m < n)(m \in A_{s+1} \backslash A_s)). \tag{1.4}$$

In this case, the approximation (A_n) always puts a smaller number m into the set A, whenever a larger number $n > m$ is deleted from A at some stage. This is a natural generalization of sets like $B := A \oplus \overline{A}$, for any c.e. set A for which $2n$ is enumerated into B whenever $2n+1$ is removed from B. Thus, according to (1.4), the number 0 can never be removed from A, the number 1 can be removed at most once, and so on. In general, the number n can be removed at most $2^n - 1$ times (and can be added at most 2^n times). Therefore, any strongly ω-c.e. set is $2^{(n+1)}$-c.e. and hence is also ω-c.e. On the other hand, according to Ershov's hierarchy theorem, not every ω-c.e. set is strongly ω-c.e. Strongly ω-c.e. sets are also called *almost c.e.* (see [7])

Theorem 1.3.1 (Soare [37], Ambos-Spies et al. [1], Calude et al. [4]). *A real number x is c.e. iff its binary expansion is a strongly ω-c.e. set.*

Obviously, the sum of two c.e. numbers is also c.e., i.e., the class **CE** is closed under addition. If x is c.e., then $-x$ is co-c.e. but not c.e. if x is not computable. Therefore, we should consider the class **SC** of semi-computable real numbers if we are interested in the algebraic properties related to c.e. real numbers. Unfortunately, even the class **SC** does not have nice algebraic properties according to the following necessary condition of semi-computable real numbers. Remember that, for any sets of natural numbers A and B, the set A is *Turing reducible* to the set B (denoted by

$A \leq_T B$) iff there is a Turing machine M such that M can compute A with B as an oracle, i.e., $M^B(n) = A(n)$ for all n.

Theorem 1.3.2 (Ambos-Spies, Weihrauch and Zheng [1]). *For any c.e. sets $A, B \subseteq \mathbb{N}$, if the real number $x_{A \oplus \overline{B}}$ is semi-computable, then either $A \leq_T B$ or $B \leq_T A$, i.e., the sets A and B must be Turing comparable.*

Let A and B be two Turing incomparable c.e. sets. Then, by Theorem 1.3.2, the real number $x_{A \oplus \overline{B}} = (x_{2A} + 1/3) - x_{2B+1}$ is the difference of two c.e. real numbers $x_{2A} + 1/3$ and x_{2B+1}, but it is not semi-computable. This means that the class **SC** is not closed under the addition and subtraction operations. This leads naturally to the class of d.c.e real numbers, where a real number is called *d.c.e* (for difference of c.e. real numbers) if it is the difference of two c.e. real numbers. The class of all d.c.e. real numbers is denoted by **DCE**. Thus, the class **DCE** is a proper superset of **SC**. A stronger result related to that has been proved by Barmpalias and Lewis-Pye in [3] that for any non-computable c.e. real number x there is a c.e. real number y such that $x - y$ is not semi-computable.

The class of d.c.e. real numbers is defined algebraically as an extension of the c.e. real numbers. Computationally, d.c.e. real numbers can be described by the weakly convergent computable sequences of rational numbers. Because of the effectively converging condition $(\forall s)(|x_s - x_{s+1}| \leq 2^{-s})$, an effectively convergent sequence can make only a "small jump" in every step. Instead of exact control of the jump of every step of the approximation, we can introduce a weaker version of effective convergence by controlling the overall sum of jumps. More precisely, we say that a sequence (x_s) converges *weakly effectively* if the sum $\sum |x_s - x_{s+1}|$ is finite. Weakly effective convergence characterizes perfectly the class of d.c.e. real numbers as follows.

Theorem 1.3.3 (Ambos-Spies, Weihrauch and Zheng [1]). *A real number x is d.c.e. iff there is a computable sequence (x_s) of rational numbers which converges to x weakly effectively, i.e., $\sum |x_s - x_{s+1}| \leq c$ for some constant c.*

Proof. Let $x = y - z$ be a d.c.e. real number and let (y_s) and (z_s) be two computable increasing sequences of rational numbers which converge to y and z respectively. Consider the computable sequence (x_s) defined by $x_s := y_s - z_s$. Then the jumps $|x_s - x_{s+1}|$ are contributed either by the increasing part of $y_{s+1} - y_s$ or the decreasing part of $z_{s+1} - z_s$. Therefore $\sum |x_s - x_{s+1}| \leq (y - y_0) + (z - z_0)$. That is, the sequence (x_s) converges to x weakly effectively.

On the other hand, let (x_s) be a computable sequence of rational numbers which converges to x weakly effectively, i.e., $\sum |x_{s+1} - x_s|$ is bounded. Define two computable sequences (y_s) and (z_s) of rational numbers by

$$y_s := x_0 + \sum_{i=0}^{s} (x_{i+1} \dot{-} x_i) \quad \text{and} \quad z_s := \sum_{i=0}^{s} (x_i \dot{-} x_{i+1}).$$

where the arithmetical difference $x \dot{-} y$ is defined as $x - y$ if $x \geq y$ and as 0 otherwise. Obviously, both sequences (y_s) and (z_s) are nondecreasing and bounded. Hence

$y := \lim y_s$ and $z := \lim z_s$ exist and they are both c.e. real numbers. Furthermore,

$$y - z = \lim_{s\to\infty} y_s - \lim_{s\to\infty} z_s = \lim_{s\to\infty}\left(x_0 + \sum_{i=0}^{s}(x_{i+1} \dot{-} x_i) - \sum_{i=0}^{s}(x_i \dot{-} x_{i+1})\right)$$
$$= \lim_{s\to\infty}\left(x_0 + \sum_{i=0}^{s}(x_{i+1} - x_i)\right) = \lim_{s\to\infty} x_s = x$$

That is, x is a d.c.e. real number. □

It is not difficult to see that, if two sequences (x_s) and (y_s) converge weakly effectively to x and y, respectively, then the sequences $(x_s + y_s)$, $(x_s - y_s)$, $(x_s y_s)$ and (x_s/y_s) (suppose that $y \neq 0$ and $y_s \neq 0$ for all s) converge also weakly effectively to $x+y, x-y, xy$ and x/y, respectively. Thus, by Theorem 1.3.3, it is easy to see that the class of d.c.e. real numbers is closed under the arithmetical operations addition, subtraction, multiplication and division. That is, **DCE** is a *field*. As pointed out by Downey and Laforte in [9], and proved independently by Raichev [27] and Ng [24], **DCE** is also a real closed field. Thus, the class **DCE** is the first extension of the computable real number class which has natural computational characterizations as well as nice algebraic properties.

Next, we will ask whether the class **DCE** is closed under more general computable operations, where computable operations are understood as the computable functions. The computable real functions can be defined in many equivalent ways ([13, 25, 17, 42]). A very natural way is the definition by means of Turing machines. That is, a real function f is *computable*, iff there is an (extended, or so-called, Type-2) Turing machine which can transfer any effectively convergent sequence of rational numbers in the sense of (1.2) which converges to $x \in \mathrm{dom}(f)$ to an effectively convergent sequence of rational numbers which converges to $f(x)$ in the sense of (1.2). Any computable real function is continuous and hence the computable real functions are effectivizations of continuous real functions. It is shown in [34] that the class **DCE** is not closed under the total computable real functions. The images of d.c.e. real numbers under total computable real functions form another very interesting class. This class has a very nice computational characterization in terms of "big jumps" of an approximation.

Remember that any computable real number x has a computable approximation (x_s) of rational numbers which converges effectively, i.e., $|x_s - x_{s+1}| \leq 2^{-s}$ for all s. So we can say that such a computable approximation does not make any big jump. If a real number can only be approximated by a computable sequence of rational numbers with big jumps, then it cannot be computable. The more big jumps the sequences have to have, the less computable its limit can be. In other words, the computable level of a real number can be determined by the convergence quality of its approximation. Counting the number of big jumps is a natural way to measure the convergence quality of an approximation. If $|x_s - x_{s+1}| > 2^{-s}$, then it has a big jump at stage s. However, a big jump can be made in many steps instead of only one step. So, more naturally, a big jump should be regarded as an index pair (i, j) such that $i, j \geq n$ and $|x_i - x_j| > 2^{-n}$.

We call a real number *divergence bounded computable* (d.b.c. for short) if it can be approximated by a computable sequence of rational numbers whose number of big jumps can be bounded by a computable function. More precisely, a real number x is d.b.c. if there exist a computable function f and a computable sequence (x_s) of rational numbers which converges to x such that the number of non-overlapping index pairs (s,t) with $s,t \geq n$ and $|x_s - x_t| > 2^{-n}$ is bounded by $f(n)$ for all n. The class of all d.b.c. real numbers is denoted by **DBC**. Interestingly, the class **DBC** is exactly the image of **DCE** under total computable real functions.

Theorem 1.3.4 (Zheng et al. [34, 51, 46]). *A real number x is d.b.c. iff there is a d.c.e. real number u and a total computable real function f such that $x = f(u)$.*

The class **DBC** is another class of real numbers after **DCE** which can be characterized mathematically as well as computationally. It is not only a field, it is even closed under total computable real functions.

Finally, we call a real number x *computably approximable* (c.a. for short) if there is a computable sequence (x_s) of rational numbers which converges to x, and we denote the class of all c.a. real numbers by **CA**. Thus we get a *finite hierarchy* $\mathbf{EC} \subsetneq \mathbf{CE} \subsetneq \mathbf{DCE} \subsetneq \mathbf{DBC} \subsetneq \mathbf{CA}$, where the last inequality follows from a stronger result, Theorem 1.6.3, of Section 1.6. All these five classes are introduced in a very natural way and they all have nice mathematical and computational properties.

1.4 Differences of C.E. Real Numbers

In Section 1.3 we have seen that the class **EC** of computable real numbers is extended to **DCE** (difference of c.e.), **DBC** (divergence bounded computable) and finally **CA** (computably approximable) real numbers. This forms a finite hierarchy and each class is defined in a very natural way. In this section we are going to look at the class **DCE** more closely and we will show more interesting properties about the d.c.e real numbers.

Let's start with another equivalent characterization of d.c.e real numbers by another weakly effective convergence. While the effective convergence condition $|x_s - x_{s+1}| \leq 2^{-s}$ can be replaced by $|x - x_s| \leq 2^{-s}$ and we still get the same class of computable real numbers, the weak convergence condition $\sum_{s\in\mathbb{N}} |x_s - x_{s+1}| \leq c$ cannot be equivalently replaced by, say, $\sum_{s\in\mathbb{N}} |x - x_s| \leq c$ for a constant c as shown by the next proposition.

Proposition 1.4.1. *If x is the limit of a computable sequence (x_s) of rational numbers such that $\sum_{s\in\mathbb{N}} |x - x_s| \leq c$ for some constant c, then x is computable.*

Proof. Let (x_s) be a computable sequence of rational numbers which converges to x such that $\sum_{s\in\mathbb{N}} |x - x_s| \leq c$ for a constant c. W.l.o.g., we suppose that c is a rational number. Now we construct a computable sequence (y_s) of rational numbers which converges to x effectively, i.e., $|x - y_s| \leq 2^{-s}$ for all s. For any natural number n, we choose a rational number y_n such that there are at least $c2^{n+1}$ elements x_t of the

sequence (x_s) with the property that $|x_t - y_n| \leq 2^{-(n+1)}$. Because (x_s) is a convergent computable sequence, such a y_n exists and can be effectively found.

Suppose, by contradiction, that $|x - y_n| > 2^{-n}$. Then, all $c2^{n+1}$ elements x_t with the property $|x_t - y_n| \leq 2^{-(n+1)}$ have at least a distance $2^{-(n+1)}$ from x. This implies that $\sum |x - x_s| > (c2^{n+1}) \cdot (2^{-(n+1)}) = c$, a contradiction.

Therefore, x must be computable. □

Notice that, if $|x - x_s| \leq c_s$ and (c_s) is a computable sequence of rational numbers which converges to 0, then the limit x is computable. However, if we consider a computable sequence (c_s) of c.e. real numbers converging to 0, we get another equivalent characterization of the class of d.c.e. real numbers.

Definition 1.4.2. A sequence (x_s) of real numbers converges to x *computably enumerably bounded* (c.e. bounded) if there is a computable sequence (δ_s) of positive rational numbers such that $\sum_{s\in\mathbb{N}} \delta_s$ is finite and

$$(\forall s \in \mathbb{N}) \left(|x - x_s| \leq \sum\nolimits_{i \geq s} \delta_i \right). \tag{1.5}$$

Notice that, if a sequence (x_s) converges effectively to x, then $|x - x_s| \leq 2^{-s}$ holds for all s. Therefore, it is natural to say that (x_s) converges *computably bounded* if (x_s) converges effectively. For a c.e. bounded convergent sequence (x_s), a computable bound for $|x - x_s|$ which is convergent to zero is not always available. The bounds $\sum_{i\geq s} \delta_i$ in (1.5) are only c.e. real numbers which converge to 0 monotonically when s increases. The c.e. bounded convergence supplies another characterization of the class of d.c.e. real numbers as follows.

Theorem 1.4.3 (Rettinger and Zheng [32]). *A real number x is d.c.e. if and only if there is a computable sequence (x_s) of rational numbers which converges to x c.e. bounded.*

Proof. Let $x = y - z$ be a d.c.e. real number where y and z are c.e. real numbers. That is, there are increasing computable sequences (y_s) and (z_s) of rational numbers which converge to y and z, respectively. Define $x_s := y_s - z_s$ for all s. Then (x_s) is a computable sequence of rational numbers which satisfies

$$\begin{aligned} |x - x_s| &= |(y - z) - (y_s - z_s)| \leq (y - y_s) + (z - z_s) \\ &= \sum\nolimits_{i \geq s} ((y_{i+1} - y_i) + (z_{i+1} - z_i)) \end{aligned}$$

for all s. Thus, the sequence (x_s) converges c.e. bounded with respect to the sequence (δ_s) defined by $\delta_s := (y_{s+1} - y_s) + (z_{s+1} - z_s)$ for all s.

On the other hand, let (x_s) and (δ_s) (for $\delta_s > 0$) be computable sequences of rational numbers which satisfy condition (1.5). Define a computable sequence (x'_s) inductively by $x'_0 := x_0$ and

$$x'_{s+1} := \begin{cases} x_{s+1} & \text{if } |x'_s - x_{s+1}| \leq \delta_s; \\ x'_s + \delta_s & \text{if } |x'_s - x_{s+1}| > \delta_s \text{ and } x'_s < x_{s+1}; \\ x'_s - \delta_s & \text{if } |x'_s - x_{s+1}| > \delta_s \text{ and } x'_s > x_{s+1}, \end{cases} \tag{1.6}$$

for all s. By induction we can prove that the sequence (x'_s) satisfies the following condition:

$$(\forall s \in \mathbb{N}) \left(|x'_s - x'_{s+1}| \leq \delta_s \ \& \ |x - x'_{s+1}| \leq \sum_{i>s} \delta_i \right). \tag{1.7}$$

The second inequality of (1.7) implies that the sequence (x'_s) converges to x. The first inequality of (1.7) implies furthermore that this convergence is weakly effective, i.e., $\sum_{s\in\mathbb{N}} |x'_s - x'_{s+1}| \leq \sum_{s\in\mathbb{N}} \delta_s$. Thus, x is a d.c.e. real number. □

Next we will investigate the randomness of the d.c.e real numbers. For more details about randomness the reader is referred to the book of Downey and Hirschfeldt [7]. We call a Turing machine *prefix-free* if its domain is a prefix-free set. We define the *prefix-free complexity* of a binary string σ as $K(\sigma) := \min\{|\tau| : U(\tau) = \sigma\}$, where U is a universal prefix-free machine. A binary sequence α is called *random* if there is a constant c such that $K(\alpha \upharpoonright n) \geq n - c$ for all n (see, e.g., [20, 5]). Thus, random sequences have the largest prefix-free complexity.

A real number is *random* if its binary expansion is a random sequence. Solovay [39] has shown that a real x is random iff x is in only finitely many U_i for any computable collection $\{U_n : n \in \mathbb{N}\}$ of c.e. open sets such that $\sum_{n\in\mathbb{N}} \mu(U_n) < \infty$, where $\mu(U)$ is the Lebesgue measure of U.

As we have seen in the last section, a d.c.e. real number may be neither c.e. nor co-c.e. It is completely different for the random real numbers.

Theorem 1.4.4 (Rettinger and Zheng [32]). *If a real number x is d.c.e and random, then it is either c.e. or co-c.e., i.e., x must be semi-computable.*

Proof. Let x be a d.c.e. random real and let (x_s) be a computable sequence of rational numbers which converges to x weakly effectively, i.e., $\sum_{s\in\mathbb{N}} |x_{s+1} - x_s| < c$ for a constant c.

Assume, by contradiction, that x is neither c.e. nor co-c.e. Then there are infinitely many s such that $x_s < x$ and there are infinitely many s such that $x_s > x$ too. Thus, if $x_s < x$ for some $s \in \mathbb{N}$, then there is a $t \geq s$ such that $x_t < x < x_{t+1}$ and if $x_s > x$ then there is a $t \geq s$ such that $x_t > x > x_{t+1}$. This implies further that there are infinitely many s such that $x_s < x < x_{s+1}$. Let $U_s := (x_s, x_{s+1})$ if $x_s < x_{s+1}$ and $U_s := \emptyset$ otherwise. Then $\{U_n : n \in \mathbb{N}\}$ is a computable collection of c.e. open sets such that $\sum_{n\in\mathbb{N}} \mu(U_n) = \sum_{s\in\mathbb{N}} (x_{s+1} \dot{-} x_s) \leq \sum_{s\in\mathbb{N}} |x_{s+1} - x_s| \leq c$. Since x belongs to infinitely many U_n, x is not random. This contradicts our assumption. □

In order to compare the relative randomness of the c.e. real numbers, Solovay [39] introduced a reducibility as follows: a c.e. real number x is *Solovay reducible* to another c.e. real number y (denoted by $x \leq_S^0 y$) if there are a constant c and two increasing computable sequences (x_s) and (y_s) of rational numbers which converge to x and y, respectively, such that

$$(\forall s \in \mathbb{N})(x - x_s \leq c(y - y_s)). \tag{1.8}$$

In other words, $x \leq_S^0 y$ means that y cannot be increasingly approximated faster than x. In addition, if x is Solovay reducible to y, then we have $K(x \upharpoonright n) \leq K(y \upharpoonright n) + c$ for

some constant c and all natural numbers n (this is the so-called "Solovay property"). Therefore, that x is Solovay reducible to y implies that x has "less randomness" than y.

Solovay reducibility classifies relative randomness of c.e. real numbers quite well. Since Solovay reducibility is reflexive and transitive, the induced equivalence classes are defined as Solovay degrees (S-degrees). Downey, Hirschfeldt and Nies [8] showed that the S-degrees of all c.e. real numbers form a dense distributive upper semilattice, where the natural join operation is induced simply by addition. As expected, the smallest S-degree consists of all computable real numbers. More interestingly, the class of all c.e. random real numbers forms the largest S-degree among the c.e. S-degrees. This is proven by a series of works. At first, Chaitin [5] introduced Ω *reals* as halting probabilities of universal prefix-free Turing machines, i.e., $\Omega_U := \sum_{U(\sigma)\downarrow} 2^{-|\sigma|}$ where U is a prefix-free universal Turing machine. Chaitin showed that every Ω real is c.e. random. Solovay [39] called a c.e. real Ω*-like* if it is complete under the Solovay reducibility on the c.e. real numbers, i.e., every c.e. real is S-reducible to it. Then he showed that every Ω real is Ω-like and every Ω-like real is c.e. random. Next, Calude, Hertling, Khoussainov and Wang [4], showed that every Ω-like real is actually an Ω real. Finally, Kuçera and Slaman [19] closed this circle by showing that any c.e. random real is Ω-like. Therefore, the notions Ω real, Ω-like real and c.e. random real are equivalent. We will see that the random c.e. real numbers are closely related to d.c.e. real numbers too. For this purpose, we should first extend the Solovay reducibility to a larger class than **CE**.

The straightforward extension of Solovay reducibility to the c.a. real numbers does not look very natural. For example, if we extend Solovay reducibility to the class **CA** of all computably approximable real numbers simply using the inequality $|x-x_s| \leq c|y-y_s|$ instead of (1.8), then this new reduction is not even transitive on the class **DCE**. In fact, it is shown in [49] that, for any computable real number x, if x is not a rational number, then there is a d.c.e. real number y such that x is not reducible to y in this straightforwardly extended reducibility.

To solve this problem, Rettinger and Zheng [49] introduced another extension of Solovay reducibility on the c.a. real numbers in the following way: A c.a. real number x is *Solovay reducible* to another c.a. real number y (denoted by $x \leq_S y$) if there are a constant c and two (not necessarily increasing) computable sequences (x_s) and (y_s) of rational numbers which converge to x and y, respectively, such that

$$(\forall s \in \mathbb{N})(|x-x_s| \leq c(|y-y_s| + 2^{-s})). \tag{1.9}$$

As shown in [49], this *extended Solovay reducibility* is reflexive, transitive and hence is a reasonable reducibility. It has the Solovay property and coincides with the original Solovay reducibility on c.e. real numbers as well.

This new version of Solovay reduction is very closely related to the class of d.c.e. real numbers. It is easy to see that, if y is d.c.e. and $x \leq_S y$, then x is d.c.e. too. That is, the class **DCE** is closed downward under this reduction. More interestingly, we will see that the Ω-numbers are Solovay complete for the class **DCE**. Since **DCE** is the smallest field containing all c.e. real numbers, it suffices to show that

the class of real numbers which are Solovay reducible to an Ω-number is a field too. For any c.a. real number x, let $S(\leq x)$ denote the class of all real numbers which are Solovay reducible to x, i.e., $S(\leq x) := \{y \in \mathbb{R} : y \leq_S x\}$. We first show a more general result that $S(\leq x)$ is a field for any c.a. real x.

We say that a real function $f : \mathbb{R}^n \to \mathbb{R}$ is *locally Lipschitz* if for each $\mathbf{x} \in \mathrm{dom}(f)$ there is a neighborhood U of $\mathbf{x}$ and a constant L such that

$$(\forall \mathbf{u}, \mathbf{v} \in U)(|f(\mathbf{u}) - f(\mathbf{v})| \leq L \cdot |\mathbf{u} - \mathbf{v}|), \tag{1.10}$$

where $|\mathbf{u} - \mathbf{v}| := \sum_{i=1}^{n} |u_i - v_i|$ for $\mathbf{u} = (u_1, \ldots, u_n)$ and $\mathbf{v} = (v_1, \ldots, v_n)$.

Theorem 1.4.5 (Rettinger and Zheng [32]). *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a locally Lipschitz computable function and let d be a computably approximable real number. Then the class $S(\leq d)$ is closed under the function f.*

Proof. We prove only the case for $n = 2$ here. Let d be the limit of a computable sequence (d_s) of rational numbers, and let $x, y \in S(\leq d)$. Then there are computable sequences (x_s) and (y_s) of rational numbers which converge to x and y, respectively, and a common constant c such that

$$|x - x_s| \leq c(|d - d_s| + 2^{-s}) \text{ and } |y - y_s| \leq c(|d - d_s| + 2^{-s})$$

for all $s \in \mathbb{N}$.

Let L be a Lipschitz constant which satisfies condition (1.10). Assume, w.l.o.g., that all (x_s, y_s) are located in the neighborhood U in which the condition (1.10) holds. By the sequential computability of the computable function f (see, e.g., [25]), the sequence (z_s) defined by $z_s := f(x_s, y_s)$ for all s is a computable sequence of real numbers. Furthermore, this sequence satisfies the condition

$$|f(x,y) - z_s| = |f(x,y) - f(x_s, y_s)| \leq L(|x - x_s| + |y - y_s|) \leq 2cL(|d - d_s| + 2^{-s})$$

for all $s \in \mathbb{N}$. This implies that $f(x,y) \leq_S d$ and hence $f(x,y) \in S(\leq d)$. That is, the class $S(\leq d)$ is closed under the function f. □

Since the addition, subtraction, multiplication and division functions are locally Lipschitz functions, we have the following result immediately.

Corollary 1.4.6. *For any computably approximable real y, the class $S(\leq y)$ is a field.*

This immediately implies the following theorem.

Theorem 1.4.7 (Rettinger and Zheng [32]). *If y is an Ω-number, then $S(\leq y) =$ **DCE**. Therefore, a real number is d.c.e. iff it is Solovay reducible to an Ω-number.*

Random numbers have another interesting property concerning convergence of computable sequences: In [2] it was shown that for computable sequences (x_s) and (y_s) converging to random c.e. numbers x and y, respectively, the rates of convergence define a number

$$\frac{\partial x}{\partial y} = \lim_{s\to\infty} \frac{x - x_s}{y - y_s}$$

which is independent of the choice of the sequences but only depends on x and y. Furthermore it was shown that the difference $x - y$ is random if and only if this fraction is not 1. This beautiful result was extended by Joseph S. Miller in [21] to d.c.e. numbers. To this end let Ω be a fixed Ω-number and define the derivative ∂x of a d.c.e. number x by

$$\partial x = \frac{\partial x}{\partial \Omega} = \lim_{s\to\infty} \frac{x - x_s}{\Omega - \omega_s}$$

where (x_s) and (ω_s) are computable sequences converging to x and Ω, respectively.

Based on this definition, Miller proved the following statement.

Theorem 1.4.8 (Miller [21]).

1. *∂x does not depend on the choice of (x_s) and (ω_s)*
2. *$\partial x > 0$ iff x is a random left-c.e. number*
3. *$\partial x < 0$ iff x is a random right-c.e. number*

One implication of this result is that the class of non-random d.c.e. numbers forms a field.

1.5 Divergence Bounded Computable Real Numbers

As mentioned in Section 1.3, the divergence bounded computable real numbers can be characterized as the images of d.c.e. real numbers under total computable functions and the class **DBC** is actually closed under total computable real functions. Equivalently, a real number is d.b.c. if there exists a computable sequence of rational numbers which converges to it with computably bounded numbers of big jumps. The name of "divergence bounded computable" comes from this second condition.

More precisely, we have the following definition.

Definition 1.5.1 (Rettinger et al. [34] and Zheng et al. [47]). Let $h : \mathbb{N} \to \mathbb{N}$ be a total function, x be a real number and C be a class of total functions $f : \mathbb{N} \to \mathbb{N}$.

1. A sequence (x_s) of rational numbers converges *h-bounded effectively* if there are at most $h(n)$ non-overlapping index pairs (i, j) such that $|x_i - x_j| \geq 2^{-n}$, where two pairs (i_1, j_1) and (i_2, j_2) of indices are *non-overlapping* if either $i_1 < j_1 \leq i_2 < j_2$ or $i_2 < j_2 \leq i_1 < j_1$.
2. A real number x is *h-bounded computable* (h-b.c. for short) if there is a computable sequence (x_s) of rational numbers which converges to x h-bounded effectively.
3. A real number x is C-bounded computable (C-b.c. for short) if it is h-b.c. for some function h in C.

In particular, if C is the class of computable functions, then the C-b.c. real numbers are called *divergence bounded computable* (*d.b.c.* for short). The classes of h-b.c. real numbers and C-b.c. real numbers are denoted by h-**BC** and C-**BC**, respectively.

If we count the jumps of a size between 2^{-n} and 2^{-n+1} which appear after stage n, then we will have the f-Cauchy computability which is investigated in [48]. The f-Cauchy computability has a very nice hierarchy theorem that, for any computable functions f and g, if there are infinitely many n such that $f(n) < g(n)$, then there is a g-Cauchy computable real number which is not f-Cauchy computable. Therefore, we can have a hierarchy similar to the Ershov hierarchy. However, the real number classes defined by f-Cauchy computability usually do not have good mathematical properties. For example, the class of all f-Cauchy computable real numbers for constant functions f is not closed under addition.

Notice that, if we call an index pair (i, j) with $|x_i - x_j| \geq 2^{-n}$ a 2^{-n}-*jump* of the sequence (x_s) of rational numbers, then, for any 2^{-n}-jump (i, j), the binary expansions of x_i and x_j differ at their first n positions. As a result, any real numbers of an ω-c.e. binary expansion must be divergence bounded computable. However, not every d.b.c. real number has an ω-c.e. binary expansion (see Theorem 1.6.3 and Theorem 1.6.2 of Section 1.6). Let ω-b**CE** be the class of real numbers with ω-c.e. binary expansions. Then we have ω-b**CE** $\subsetneq$ **DBC**. On the other hand, it follows from Theorem 1.3.4 that we have **DCE** $\subseteq$ **DBC**. In order to show that this inclusion is proper, it suffices to show that ω-b**CE** is not a subset of **DCE**.

Theorem 1.5.2 (Ambos-Spies, Weihrauch and Zheng [1]). *For any set $A \subseteq \mathbb{N}$, if x is a d.c.e. real number of the binary expansion $2A$, i.e., $x = x_{2A}$, then the set A must be h-c.e. for the function $h = \lambda n.2^{3n}$.*

Because there exists an ω-c.e. set A which is not 2^{3n}-c.e. the real number x_{2A} has an ω-c.e. binary expansion, hence it is d.b.c., but it is not d.c.e. Therefore, we have ω-b**CE** $\not\subseteq$ **DCE** which implies that **DCE** $\subsetneq$ **DBC**. Actually, by Theorem 1.6.1, not every d.c.e. real number has an ω-c.e. binary expansion. So the classes ω-b**CE** and **DCE** are incomparable. By a direct construction proof (see [34]) or by a stronger result (see Theorem 1.6.3 of Section 1.6), there exist computably approximable real numbers which are not d.b.c. Thus, we have the separation results **DCE** $\subsetneq$ **DBC** $\subsetneq$ **CA**. In Section 1.6, we will show a stronger separation theorem among the classes **DCE**, **DBC** and **CA** (Theorem 1.6.3).

Now we are going to investigate more properties of the classes of f-**BC** for different functions f. Naturally, we are only interested in nondecreasing functions f. First, from Definition 1.5.1, it is not difficult to see that, if $\liminf f(n)$ is finite, then an f-b.c. real number must be rational, and if $f(n) \leq g(n)$ for almost all natural umbers n, then f-**BC** $\subseteq$ g-**BC**. On the other hand, if the distance between the functions f and g is bounded by a constant c, i.e., $|f(n) - g(n)| \leq c$, then we have f-**BC** $=$ g-**BC** (cf. [47]). However, if the distance between the functions f and g is unbounded, then f-**BC** is different from g-**BC**.

Theorem 1.5.3 (Zheng et al. [47]). *Let $f, g : \mathbb{N} \to \mathbb{N}$ be two computable functions such that*

$$(\forall c \in \mathbb{N})(\exists m \in \mathbb{N})(c + f(m) < g(m)).$$

*Then there exists a g-b.c. real which is not f-b.c., i.e., g-**BC** $\not\subseteq$ f-**BC**.*

Proof. We will construct a computable sequence (x_s) of rational numbers which converges g-bounded effectively to a non-f-b.c. real number x. That is, x satisfies, for all $e \in \mathbb{N}$, the following requirements

$$R_e: \quad (\varphi_e(s))_{s\in\mathbb{N}} \text{ converges } f\text{-bounded effectively to } y_e \Longrightarrow y_e \neq x,$$

where (φ_e) is an effective enumeration of the partial computable functions $\varphi_e :\subseteq \mathbb{N} \to \mathbb{Q}$. The idea to satisfy a single requirement R_e is easy. We choose an interval I and a natural number m such that $f(m) < g(m)$. Choose further two subintervals $I_e, J_e \subset I$ such that I_e and J_e have at least a distance 2^{-m}. Then we can find a real x either from I_e or J_e to avoid the limit y_e of the sequence $(\varphi_e(s))$. To satisfy all the requirements simultaneously, we use a finite injury priority construction. In the following construction, we use a second index s to denote the parameters constructed up to stage s. For example, $I_{e,s}$ denotes the current value of I_e at stage s; and $\varphi_{e,s}(n) = m$ means that the Turing machine M_e which computes φ_e outputs m in s steps with the input n. However, if it is clear from the context, we often drop the extra index s.

The formal construction of the sequence (x_s):

Stage $s = 0$: We take the unit interval $[0;1]$ as the base interval for R_0 and let $I_0 := [2^{-(m_0+1)}; 2 \cdot 2^{-(m_0+1)}]$, $J_0 := [4 \cdot 2^{-(m_0+1)}; 5 \cdot 2^{-(m_0+1)}]$ where $m_0 := \min\{m : m \geq 3 \ \&\ f(m) < g(m)\}$. Then define $x_0 := 3 \cdot 2^{-(m_0+2)}$. Notice that, the intervals I_0 and J_0 have the same length $2^{-(m_0+1)}$ and the distance between them is 2^{-m_0}. The rational number x_0 is the middle point of I_0. We need another parameter t_e to denote that $\varphi_e(t_e)$ is already used for our strategy. At this stage, let $t_{e,0} := -1$ for all $e \in \mathbb{N}$.

Stage $s+1$: Given $t_{e,s}$, x_s and the rational intervals $I_0, I_1, \cdots, I_{k_s}$ and $J_0, \cdots J_{k_s}$ for some $k_s \geq 0$ such that $I_e, J_e \subsetneq I_{e-1}$, $l(I_e) = l(J_e) = 2^{-(m_e+1)}$ and the distance between the intervals I_e and J_e is also 2^{-m_e}, for all $0 \leq e \leq k_s$. We say that a requirement R_e *requires attention* if $e \leq k_s$ and there is a natural number $t > t_{e,s}$ such that $\varphi_{e,s}(t) \in I_{e,s}$ and φ_e has not made more than $f(m_e)$ jumps of distances larger than 2^{-m_e} so far. That is, $\max G_{e,s}(m_e, t) \leq f(m_e)$, where $G_{e,s}(n,t)$ denotes the following finite set

$$\{m : (\exists i_1 < j_1 \leq \cdots i_m < j_m)(\forall d < m)(|\varphi_{e,s}(i_d) - \varphi_{e,s}(j_d)| \geq 2^{-n})\}.$$

Let R_e be the requirement of the highest priority (i.e., of the minimal index) which requires attention and let t be the corresponding natural number. Then we exchange the intervals I_e and J_e, that is, define $I_{e,s+1} := J_{e,s}$ and $J_{e,s+1} := I_{e,s}$. All intervals I_i and J_i for $i > e$ are set to be undefined. Besides, define $x_{s+1} := \mathrm{mid}(I_{e,s+1})$, $t_{e,s+1} := t$ and $k_{s+1} := e$. In this case, we say that R_e *receives attention* and the requirements R_i for $e < i \leq k_s$ are *injured* at this stage.

Otherwise, suppose that no requirement requires attention at this stage. Let $e := k_s$ and let n_s be the maximal $m_{i,t}$ which were defined so far for some $i \in \mathbb{N}$ and $t \leq s$. Denote by $j(s)$ the number of non-overlapping index pairs (i, j) such that $i < j \leq s$ and $|x_i - x_j| \geq 2^{-n_s}$. Then define

$$m_{e+1} := (\mu m)\,(m \geq n_s + 3 \;\&\; j(s) + f(m) < g(m)). \tag{1.11}$$

Choose five rational numbers a_i (for $i \leq 4$) by $a_0 := x_s - 2^{-(m_{e+1}+2)}$ and $a_i := a_0 + i \cdot 2^{-(m_{e+1}+1)}$ for $i := 1,2,3,4$. Then define the intervals $I_{e+1,s+1} := [a_0; a_1]$, $J_{e+1,s+1} := [a_3; a_4]$ and let $x_{s+1} := x_s$. Notice that the intervals I_{e+1} and J_{e+1} have length $2^{-(m_{e+1}+1)}$ and the distance between them is $2^{-m_{e+1}}$. Furthermore, x_{s+1} is the middle point of both intervals I_e and I_{e+1}.

This ends the formal construction. To show that our construction succeeds, it suffices to prove the following claims.

Claim 1.5.3.1 *For any $e \in \mathbb{N}$, the requirement R_e requires and receives attention only finitely many times.*

Proof. By induction hypothesis we suppose that there is a stage s_0 such that no requirement R_i for $i < e$ receives attention after stage s_0. Then $m_{e,s} = m_{e,s_0}$ for all $s \geq s_0$. The intervals I_e and J_e may be exchanged after stage s_0 if R_e receives attention. Notice that, if R_e receives attention at stages $s_2 > s_1 (> s_0)$ successively, then we have $|\varphi_e(t_{e,s_1}) - \varphi_e(t_{e,s_2})| \geq 2^{-m_{e,s_0}}$, because the distance between the intervals I_e and J_e is $2^{-m_{e,s_0}}$. This implies that R_e can receive attention after stage s_0 at most $f(m_{e,s_0}) + 1$ times because of the condition $\max G_{e,s}(m_e, t) \leq f(m_e)$ and hence R_e receives attention finitely often totally. □ (claim)

Claim 1.5.3.2 *The sequence (x_s) converges g-bounded effectively to some x and hence x is g-bounded computable.*

Proof. By the construction, if $x_s \neq x_{s+1}$, then there is an e such that R_e receives attention at stage $s+1$. In this case, we have $2^{-m_{e,s}} < |x_s - x_{s+1}| < 2^{-m_{e,s}+1}$. In addition, if R_e receives attention according to the same $m_{e,s}$ at stage $s+1$ and $t+1 (> s+1)$ consecutively, then we have $|x_s - x_{t+1}| \leq 2^{-(m_{e,s}+1)}$ again because of $l(I_{e,s}) = 2^{-(m_{e,s}+1)}$. This means that, if a natural number n is never chosen as $m_{e,s}$ for some e at some stage s, then there are no stages s_1, s_2 such that $2^{-n} \leq |x_{s_1} - x_{s_2}| \leq 2^{-n+1}$. Therefore, it suffices to show that, for any $m_{e,s}$, there are at most $g(m_{e,s})$ non-overlapping index pairs (i, j) such that $|x_i - x_j| \geq 2^{-m_{e,s}}$.

Given any $m_{e,s}$, suppose that it is defined for the first time at stage s according to condition (1.11). Then, there are only $j(s)$ non-overlapping index pairs (i, j) such that $|x_i - x_j| \geq 2^{-m_{e,s}}$ up to stage s. After stage s, each such jump corresponds to a stage at which R_e receives attention according to the $m_{e,s}$. However, R_e can receive attention at most $f(m_{e,s}) + 1$ times according to this same $m_{e,s}$ and $j(s) + f(m_{e,s}) < g(m_{e,s})$. Therefore, there are at most $g(m_{e,s})$ non-overlapping jumps of (x_s) which are bigger than $2^{-m_{e,s}}$. Thus the computable sequence (x_s) converges g-bounded effectively to a g-b.c. real x. □ (claim)

Claim 1.5.3.3 *The real x satisfies all requirements R_e. Therefore, x is not f-bounded computable.*

Proof. For any $e \in \mathbb{N}$, suppose that φ_e is a total function and $(\varphi_e(s))$ converges f-bounded effectively. By Claim 1.5.3.1, we can choose an s_0 such that $k_{s_0} \geq e$ and no requirement R_i for $i \leq e$ requires attention after stage s_0. This means that $I_e := I_{e,s_0} = I_{e,s}$ and $t_e := t_{e,s_0} = t_{e,s}$ for any $s \geq s_0$. By the definition of the sequence (x_s), we have $x_s \in I_e$ for all $s \geq s_0$ and hence $x \in I_e$.

Assume by contradiction that $x = \lim_{s\to\infty} \varphi_e(s)$. Then there is a stage s and a $t > t_e$ such that $\varphi_e(v)$ is defined for all $v \leq t$ and $\varphi_e(t) \in I_e$. Since $(\varphi_e(v))$ converges f-bounded effectively, $\max G_{e,s}(m_e,t) \leq f(m_e)$. That is, R_e requires attention and will receive attention at stage $s+1$. This contradicts the choice of s_0. □ (claim)

By Claim 1.5.3.2 and Claim 1.5.3.3, the real number x is g-bounded computable but not f-bounded computable. This completes the proof of the theorem. □

Corollary 1.5.4. *If $f,g : \mathbb{N} \to \mathbb{N}$ are computable functions such that $f \in o(g)$, then f-**BC** $\subsetneq$ g-**BC**.*

The classes of f-**BC** real numbers form a very natural hierarchy. However, in order to have nice mathematical properties, we have to consider a class C of functions instead of a single function f. A very natural and simple condition on the class C guarantees that the class C-**BC** is a field as shown in the next theorem.

Theorem 1.5.5 (Zheng et al. [47]). *Let C be a class of functions which contains the successor function $S(n) := n+1$ as well as all constant functions and is closed under addition and composition. Then the class C-**BC** is a field.*

Proof. Suppose that C is a class of functions which contains the successor function and all constant functions, and suppose also that C is closed under addition and composition. For any real numbers $x, y \in C$-**BC**, there are two computable sequences (x_s) and (y_s) of rational numbers which converge f-bounded effectively and g-bounded effectively to x and y, respectively, for two functions $f, g \in C$.

Let $z_s := x_{s+1} + y_{s+1}$ for all s. Then the computable sequence (z_s) converges to $x+y$. For any natural number n and any indices s,t, if $|z_s - z_t| > 2^{-n}$ then we have either $|x_{s+1} - x_{t+1}| > 2^{-(n+1)}$ or $|y_{s+1} - y_{t+1}| > 2^{-(n+1)}$. By assumption, the number of such non-overlapping index pairs is bounded by $f(n+1) + g(n+1)$. This means that the sequence (z_s) converges h-effectively to $x+y$ for the function $h(n) := f(n+1) + g(n+1)$. Since h belongs to C, the sum $x+y$ is a C-bounded computable real number too.

Analogously, we can show that $x - y \in C$-**BC**.

For the product xy, we choose a natural number N large enough such that for all s, $\max\{|x_s|, |y_s|\} \leq 2^N$ hold. Define a computable sequence (z_s) by $z_s := x_{s+N+1}y_{s+N+1}$ for all s. Notice that if

$$|x_{s+N+1} - x_{t+N+1}| \leq 2^{-(n+N+1)} \text{ and } |y_{s+N+1} - y_{t+N+1}| \leq 2^{-(n+N+1)},$$

then we have

$$|z_s - z_t| \leq |x_{s+N+1}||y_{s+N+1} - y_{t+N+1}| + |y_{t+N+1}||x_{s+N+1} - x_{t+N+1}| \leq 2^{-n}.$$

In other words, if $|z_s - z_t| > 2^{-n}$, then we have either $|x_{s+N+1} - x_{t+N+1}| > 2^{-(n+N+1)}$ or $|y_{s+N+1} - y_{t+N+1}| > 2^{-(n+N+1)}$. This implies that the sequence (z_s) converges h-bounded effectively to xy for the function $h(n) := f(n+N+1) + g(n+N+1)$. Since the function h belongs to the class C, we have $xy \in C$-**BC**.

Finally we consider the division x/y for $y \neq 0$. Suppose without loss of generality that $y_s \neq 0$ for all s. Choose a natural number N such that $2^{-N} \leq |y_s|$ and $\max\{|x_s|, |y_s|\} \leq 2^N$ for all s. Define a computable sequence (z_s) by $z_s := x_{s+3N+1}/y_{s+3N+1}$. For any s, t and n, if we have

$$|x_{s+3N+1} - x_{t+3N+1}| \leq 2^{-(n+3N+1)} \text{ and } |y_{s+3N+1} - y_{t+3N+1}| \leq 2^{-(n+3N+1)}$$

then we have

$$\begin{aligned} |z_s - z_t| &= \left| \frac{x_{s+3N+1}}{y_{s+3N+1}} - \frac{x_{t+3N+1}}{y_{t+3N+1}} \right| \\ &\leq \frac{|x_{s+3N+1}||y_{s+3N+1} - y_{t+3N+1}| + |y_{s+3N+1}||x_{s+3N+1} - x_{t+3N+1}|}{|y_{s+3N+1} y_{t+3N+1}|} \\ &\leq 2^{-n}. \end{aligned}$$

Thus if the natural numbers s, t satisfy $|z_s - z_t| > 2^{-n}$, then we have

$$|x_{s+3N+1} - x_{t+3N+1}| > 2^{-(n+3N+1)} \text{ or } |y_{s+3N+1} - y_{t+3N+1}| > 2^{-(n+3N+1)}.$$

This implies that the computable sequence (z_s) converges h-bounded effectively to x/y for the function $h(n) := f(n+3N+1) + g(n+3N+1)$. Therefore the quotient x/y is C-bounded effectively computable since $h \in C$. □

Thus, according to Theorem 1.5.5, we can define a lot of classes of real numbers between **EC** and **CA** which have different levels of computability and they are fields as well. For example, C can be the class of computable linear functions, computable polynomial functions, primitive recursive functions, or the class of computable functions, etc. However, it is not clear yet for which class C of functions we have C-**BC** = **DCE**. Let $o(ep)$ be the class of functions f such that $\lim_{n\to\infty} f(n)/2^n = 0$, and let $o_e(ep)$ be the class of computable functions in $o(ep)$. Then it is shown in [47] that **SC** $\subseteq o(ep)$-**BC** and **SC** $\nsubseteq o_e(ep)$-**BC**.

Now we are going to explore the relationship between the class **DBC** and reducibility defined on the class **CA**. As we mentioned in Section 1.4, the class of d.c.e real numbers can be characterized by the (extended) Solovay reducibility, where a real number x is Solovay reducible to another real number y (denoted by $x \leq_S y$) if there are a constant c and two computable sequences (x_s) and (y_s) which converge to x and y respectively, such that $|x - x_s| \leq c(|y - y_s| + 2^{-s})$ for all s. We have seen that a real number is d.c.e iff it is Solovay reducible to a random c.e. real number. In some sense, Solovay reducibility is a perfect match for the class of d.c.e. real numbers. Now we are trying to find a corresponding reducibility for the class of d.b.c. real numbers.

Notice that condition (1.9) of extended Solovay reducibility requires essentially that the approximation errors of x are bounded by a fixed linear combination of the approximation errors of y. This "linear combination domination" can be replaced by a more general "computable domination" as follows.

Definition 1.5.6 (Zheng and Rettinger [50]). A c.a. real number x is *convergence-dominated reducible* (cd-reducible, for short) to another c.a. real number y (denoted by $x \leq_{cd} y$) if there exist a monotone total computable real function $h : \mathbb{R} \to \mathbb{R}$ with $h(0) = 0$ and two computable sequences (x_s) and (y_s) of rational numbers which converge to x and y, respectively, such that

$$\forall s \in \mathbb{N}(|x - x_s| \leq h(|y - y_s|) + 2^{-s}). \tag{1.12}$$

Since we are interested only in comparing convergence speeds of two converging sequences, the exact values of the function h in the inequality (1.12) are not really important. It only matters how "large" it is, because it serves only as upper bounds. This definition using a computable real function looks general enough, but a discrete version might be convenient in many related proofs.

Lemma 1.5.7. *A c.a. real number x is cd-reducible to another c.a. real number y if and only if there are two computable sequences (x_s) and (y_s) which converge to x and y, respectively, and there is a computable function $h : \mathbb{N} \to \mathbb{N}$ such that*

$$|y - y_s| \leq 2^{-h(n)} \Longrightarrow |x - x_s| \leq 2^{-n} + 2^{-s}. \tag{1.13}$$

Convergence-dominated reducibility is closely related to the class of divergence bounded computable real numbers as shown in the next theorem.

Theorem 1.5.8 (Zheng and Rettinger [50]). *For any c.a. real numbers x and y, we have*

1. *If $x \leq_S y$ then $x \leq_{cd} y$.*
2. *The class* **DBC** *is closed downward under the reduction $\leq_{cd}$. That is, if $x \leq_{cd} y$ and y is d.b.c., then x is d.b.c. too.*
3. *x is cd-reducible to an Ω-number if and only if x is d.b.c. In other words, Ω-numbers are cd-complete for the class* **DBC**.

Therefore, the Ω-numbers (i.e., the c.e. random real numbers) play the same role for the class **DBC** with respect to cd-reducibility as for the class **DCE** with respect to Solovay reducibility.

1.6 Turing Degrees of Computably Approximable Real Numbers

In the previous sections we have considered the real numbers which are limits of computable sequences of rational numbers with various restrictions on the convergences. Different levels of efficiency of convergence guarantee different levels of

computability of the limits. If we do not require any efficiency of convergence, we reach the class of computably approximable real numbers. In other words, a real number is *computably approximable* if it is the limit of a computable sequence of rational numbers. So, from a computation point of view, the class **CA** is the largest class of real numbers which can somehow be computed.

In this section, we are going to explore the comparison of the c.a. real numbers in terms of Turing reducibility. Turing reducibility is firstly defined between two sets of natural numbers. For two sets $A, B \subseteq \mathbb{N}$, we say that A is Turing reducible to B (denoted by $A \leq_T B$) if there is a Turing machine Φ such that A can be decided by Φ with the oracle B, that is, $A(n) = \Phi^B(n)$ for all n. Turing reducibility can be naturally extended to real numbers by means of binary expansion. We can call a real number x *Turing reducible* to another real number y (denoted by $x \leq_T y$ as well) if there are sets A, B such that $x = x_A, y = x_B$ and $A \leq_T B$. The *Turing degrees* and their *reducibility* can be defined straightforwardly. For example, the Turing degree of a real number x is the class of all real numbers which are Turing equivalent to x, i.e., $\deg(x) := \{y : x \equiv_T y\}$. It is not difficult to see that Turing reducibility can be defined in terms of Dedekind cuts and Cauchy sequence representations of real numbers equivalently.

Thus, the computable real numbers form a single Turing degree, denoted by $\mathbf{0}$, while c.a. real numbers have the same Turing degree structure as the Δ^0_2-Turing degrees. Among the Δ^0_2-Turing degrees, the c.e. degrees play a significantly important role in classical computability theory, where a Turing degree is called c.e. if it contains at least one c.e. set. Since every c.e. set A of natural numbers corresponds to a c.e. real number x_A of which A is its binary expansion, so every c.e. degree contains a c.e. real numbers. If we call a Turing degree $\mathbf{a}$ the *degree of a c.e. real number* if it contains at least one c.e. real number, then all c.e. degrees are degrees of a c.e. real number. On the other hand, the Dedekind cut of a c.e. real number is a c.e. set of rational numbers and hence the degree of any c.e. real number is a c.e. degree.

For the d.c.e. real numbers the situation is different. If $A = B - C$ is a d.c.e. set where B and C are c.e. sets, then the real number $x_A = x_B - x_{C \cap B}$ is obviously a d.c.e real number. This means that every d.c.e degree contains at least one d.c.e. real number, where a d.c.e. degree is a Turing degree which contains at least one d.c.e. set. The converse is not true, as shown by the following theorem, where a real number is called *strongly c.e.* if its binary expansion is a c.e. set.

Theorem 1.6.1 (Zheng [45]). *There are two strongly c.e. real numbers x and y such that the difference $x - y$ does not have an ω-c.e. Turing degree. Thus, the Turing degree of a d.c.e. real number is not necessarily ω-c.e.*

Proof. We will construct the computable sequences (A_s), (B_s) and (C_s) of finite subsets of $\mathbb{N}$ which satisfy the following conditions:

1. $x_{A_n} = x_{B_n} - x_{C_n}$ for all $n \in \mathbb{N}$.
2. The limits $A := \lim A_n$, $B := \lim B_n$ and $C := \lim C_n$ exist and $x_A = x_B - x_C$ holds.
3. (B_s) and (C_s) are computable enumerations of the c.e. sets B and C, respectively. Hence x_A is d.c.e.

4. The degree $\deg_T(A)$ is not ω-c.e.

To satisfy the third condition, we will define the finite sets B_{s+1} and C_{s+1} at stage $s+1$ in such a way that $B_s \subseteq B_{s+1}$ and $C_s \subseteq C_{s+1}$ and then define A_{s+1} by $x_{A_{s+1}} = x_{B_{s+1}} - x_{C_{s+1}}$. This automatically satisfies the first and second conditions too. For the fourth condition, it suffices to satisfy, for all ω-c.e. sets V, partial computable functionals Γ and Δ, the following requirements:

$$R_{V,\Gamma,\Delta} : A \neq \Gamma^V \text{ or } V \neq \Delta^A.$$

From an effective enumeration of all ω-c.e. sets[1] and all partial computable functionals, these requirements can also be effectively enumerated as (R_e). All requirements are given different priorities according to this enumeration. That is, R_i is of higher priority than R_j if and only if $i < j$.

The strategy for satisfying a single requirement $R_{V,\Gamma,\Delta}$ is as follows:

We choose an $x \in \mathbb{N}$ and a $y > \max\{x, \delta\gamma(x)[s]\}$, where δ, γ are use-functions of the functionals Δ and Γ (see, e.g., Soare [38]). By putting y into B or C it is possible to force x to enter or leave A because of $x_A = x_B - x_C$. For example, when $x_A = 0.\alpha 01^k$ where $|\alpha| = x$ and $k = y - x$, then by putting y (which equals $x+k$) into B, we will have $x_A = 0.\alpha 10^k$, which means that x is in A. After that, if we put y into C, then we will have $x_A = 0.\alpha 01^k$ again, which means that x is not in A anymore. The purpose of x is to witness the requirement R_e in such a way that $A(x) \neq \Gamma^V(x)$ or $V(x) \neq \Delta^A(x)$. To this end, we consider the following phases.

Phase 1: $x, y \notin A \cup B \cup C$. We wait for a stage s such that:

$$A_s(x) = \Gamma^V(x)[s] \;\&\; V \upharpoonright \gamma(x)[s] = \Delta^A \upharpoonright \gamma(x)[s]. \tag{1.14}$$

(If this never happens, then x witnesses the requirement $R_{V,\Gamma,\Delta}$ already.) Define $B_{s+1} := B_s \cup \{x\}$, $C_{s+1} := C_s$ and then enter phase 2. It can be shown that $A_{s+1} = A_s \cup \{x\}$. This implies that $A_{s+1}(x) \neq A_s(x) = \Gamma^V(x)[s]$.

Phase 2: $x \in A$ and $y \notin B \cup C$. We wait for some new stage $s' > s$ such that

$$A_{s'}(x) = \Gamma^{V\upharpoonright\gamma(x)}(x)[s'] \;\&\; V \upharpoonright \gamma(x)[s'] = \Delta^{A\upharpoonright\delta\gamma(x)} \upharpoonright \gamma(x)[s']. \tag{1.15}$$

(If this never happens, then x is also a witness of the requirement $R_{V,\Gamma,\Delta}$.) In this case, we hope to remove x from A to force the initial segment $V \upharpoonright \gamma(x)[s']$ to be changed if this condition is satisfied later again. This can be achieved by putting y into C, i.e., define $B_{s'+1} := B_{s'}$, $C_{s'+1} := C_{s'} \cup \{y\}$. Then go into phase 3.

Phase 3: $x \notin A$ and $y \in C \setminus B$. We wait for some new stage $s'' > s'$ such that

$$A_{s''}(x) = \Gamma^{V\upharpoonright\gamma(x)}(x)[s''] \;\&\; V \upharpoonright \gamma(x)[s''] = \Delta^{A\upharpoonright\delta\gamma(x)} \upharpoonright \gamma(x)[s'']. \tag{1.16}$$

[1] We do not really have an effective enumeration of all ω-c.e. sets. For our proof it suffices to consider the effective enumeration $((V_{i,s})_{s\in\mathbb{N}}, \varphi_j)_{i,j\in\mathbb{N}}$ of all computable sequences $(V_{i,s})_{s\in\mathbb{N}}$ of finite subsets $V_{i,s} \subseteq \mathbb{N}$ and partial computable functions φ_j. As long as the condition $|\{t \leq s : n \in V_{i,t} \setminus V_{i,t+1} \cup V_{i,t+1} \setminus V_{i,t}\}| \leq \varphi_j(n)$ is not destroyed at stage s, we treat $(V_{i,s})_{s\in\mathbb{N}}$ as an ω-enumeration. Otherwise, we can simply ignore the pair $((V_{i,s})_{s\in\mathbb{N}}, \varphi_j)$. However, the technical details related to this are omitted in our proof for the simplicity.

(If this never happens, then x is a witness of $R_{V,\Gamma,\Delta}$ again.) In this case we hope to put x into A again. To achieve this, we put y into B. Namely, we define $B_{s''+1} := B_{s''} \cup \{y\}$, $C_{s''+1} := C_{s''}$. Now the supplementary element y is used two times and $y \in B \cup C$. We choose a new supplementary element $y := y+1$ (this new y is not in $B \cup C$) and go to phase 2.

Notice that, if we go from phase X to phase Y, then $A(x)$, hence also $V \upharpoonright \gamma(x)$, has to be changed. But if we go from phase Y back to phase X later, $A \upharpoonright \delta\gamma(x)$ recovers the value it had in the last appearance of phase X. So, the initial segment $\Delta^{A \upharpoonright \delta\gamma(x)} \upharpoonright \gamma(x)$, hence also the initial segment $V \upharpoonright \gamma(x)$, is recovered. This can happen at most finitely often, because V is an ω-c.e. set. Therefore, after some stages, (1.15) or (1.16) will never hold again. Thus the requirement $R_{V,\Gamma,\Delta}$ is finally satisfied by the witness x.

To satisfy all the requirements, we apply a finite injury priority construction. A witness x_e and a supplementary element y_e $(> x_e)$ are appointed to the requirement R_e at any stage. Whenever an action for R_e appears, all requirements R_i with $i > e$ will be *initialized* by redefining the witnesses x_i and supplementary elements y_i of R_i such that they are bigger than all elements enumerated into B and C so far and also bigger than $\delta\gamma(x_e)$ to preserve the computation of $\Delta^{A \upharpoonright \delta\gamma(x)} \upharpoonright \gamma(x_e)$ from injury by lower priority requirements. Furthermore, we build a "firewall" between the supplementary elements y_e of R_e and the witness x_i of R_i for $i > e$. Namely, we choose a second supplementary element z_e such that $(\forall i > e)\,(y_e < z_e < x_i, y_i)$ and put it into $B \backslash C$. □

In the following, for any class $\mathbf{C}$ of sets of natural numbers and class $\mathbf{D}$ of real numbers, we will use $\mathbb{D}(\mathbf{C})$ and $\mathbb{D}_{\mathbf{R}}(\mathbf{D})$ to denote the class of degrees which contain at least one set in $\mathbf{C}$ and the class of degrees which contain at least one real number in $\mathbf{D}$. By Theorem 1.6.1, we have now $\mathbb{D}(\mathbf{DCE}) \subsetneq \mathbb{D}_{\mathbf{R}}(\mathbf{DCE})$. For the degrees of d.c.e. real numbers, we have some other results which compare with the ω-c.e. degrees and Δ^0_2-degrees as follows.

Theorem 1.6.2 (Downey, Wu and Zheng [6]).

1. *Every ω-c.e. Turing degree contains a d.c.e. real number.*
2. *There exists a Δ^0_2-Turing degree which does not contain any d.c.e. real number.*

Finally, let's look at the class **DBC** of divergence bounded computable real numbers. In some sense, the idea of defining a divergence bounded computable real number in terms of a computable sequence of rational numbers with a computably bounded number of big jumps is quite similar to the definition of ω-c.e. sets of natural numbers. However the comparison of the ω-c.e. degrees and the degrees of d.b.c. real numbers does not make sense anymore because we have seen that $\mathbb{D}(\mathbf{CE}) \subsetneq \mathbb{D}(\omega\text{-}\mathbf{CE}) \subsetneq \mathbb{D}_{\mathbf{R}}(\mathbf{DCE})$. Not very surprisingly we can expect that the class of degrees of d.b.c. real numbers is strictly between the class of degrees of d.c.e. real numbers and the class of all Δ^0_2-degrees, as the following theorem shows. Remember that a real number x is f-bounded computable (f-b.c. for short) if there is a computable sequence (x_s) of rational numbers which converges to x such that

the number of non-overlapping index pairs (i, j) with $|x_i - x_j| \geq 2^{-n}$ is bounded by $f(n)$ for all n.

Theorem 1.6.3 (Rettinger and Zheng [31, 33]).

1. *There is a Δ^0_2-Turing degree which does not contain any d.b.c. real number.*
2. *Let $f, g : \mathbb{N} \to \mathbb{N}$ be two monotonically increasing computable functions such that*

$$(\forall n)(g(n+1) \geq g(n)+2) \qquad \textit{and} \tag{1.17}$$

$$(\exists \gamma > 1)(\forall c \in \mathbb{N})(\forall^\infty n)(f(\gamma n)+n+c < g(n)) \tag{1.18}$$

   Then there is a g-b.c. real number x which is not Turing equivalent to any f-b.c. real number.
3. *There is a d.b.c. real number which is not Turing equivalent to any d.c.e real number.*

The proofs of Items 1. and 2. of Theorem 1.6.3 are relatively complicated constructions. Item 3. follows from Item 2. and the fact that any d.c.e. real number is 2^n-bounded computable (see, e.g., [44]).

Combining Theorem 1.6.1, Theorem 1.6.2 and Theorem 1.6.3 we have the following relationship of the classes of Turing degrees.

$$\mathbb{D}(\mathbf{CE}) \subsetneq \mathbb{D}(\omega\text{-}\mathbf{CE}) \subsetneq \mathbb{D}_{\mathbf{R}}(\mathbf{DCE}) \subsetneq \mathbb{D}_{\mathbf{R}}(\mathbf{DBC}) \subsetneq \mathbb{D}(\Delta^0_2).$$

Therefore, the classes of Turing degrees of real number classes discussed here form a proper hierarchy as well.

References

1. Ambos-Spies, K., Weihrauch, K., Zheng, X.: Weakly computable real numbers. Journal of Complexity **16**(4), 676–690 (2000)
2. Barmpalias, G., Lewis-Pye, A.: Differences of halting probabilities. J. Comput. Syst. Sci. **89**, 349–360 (2017)
3. Barmpalias, G., Lewis-Pye, A.: A note on the differences of computably enumerable reals. In: A. Day, M. Fellows, N. Greenberg, B. Khoussainov, A. Melnikov, F. Rosamond (eds.) Computability and Complexity: Essays Dedicated to Rodney G. Downey on the Occasion of His 60th Birthday, *Lecture Notes in Computer Science*, vol. 10010, pp. 623–632. Springer, Cham (2017)
4. Calude, C.S., Hertling, P.H., Khoussainov, B., Wang, Y.: Recursively enumerable reals and Chaitin ω numbers. Theoretical Computer Science **255**, 125–149 (2001)
5. Chaitin, G.J.: A theory of program size formally identical to information theory. Journal of the Association for Computing Machinery **22**, 329–340 (1975)
6. Downey, R., Wu, G., Zheng, X.: Degrees of d.c.e. reals. Mathematical Logic Quarterly **50**(4,5), 345–350 (2004)
7. Downey, R.G., Hirschfeldt, D.R.: Algorithmic Randomness and Complexity. *Theory and Applications of Computability*. Springer, New York (2010)
8. Downey, R.G., Hirschfeldt, D.R., Nies, A.: Randomness, computability, and density. SIAM J. Comput. **31**(4), 1169–1183 (electronic) (2002)

9. Downey, R.G., LaForte, G.L.: Presentations of computably enumerable reals. Theoretical Computer Science **284**(2), 539–555 (2002)
10. Eršov, J.L.: A certain hierarchy of sets. I (Russian). Algebra i Logika **7**(1), 47–74 (1968)
11. Eršov, J.L.: A certain hierarchy of sets. II (Russian). Algebra i Logika **7**(4), 15–47 (1968)
12. Eršov, J.L.: A certain hierarchy of sets. III (Russian). Algebra i Logika **9**, 34–51 (1970)
13. Grzegorczyk, A.: On the definitions of computable real continuous functions. Fundamenta Mathematicae **44**, 61–71 (1957)
14. Hayman, W.K., Wu, J.M.G.: Level sets of univalent functions. Comment. Math. Hel. **56**, 366–403 (1981)
15. Hertling, P., Spandl, C.: Computing a solution of Feigenbaum's functional equation in polynomial time. Logical Methods in Computer Science **10**(4), 4:7, 9 (2014)
16. Ko, K.I.: On the definitions of some complexity classes of real numbers. Math. Systems Theory **16**, 95–109 (1983)
17. Ko, K.I.: Complexity Theory of Real Functions. *Progress in Theoretical Computer Science*. Birkhäuser, Boston (1991)
18. Kreitz, C., Weihrauch, K.: Complexity theory on real numbers and functions. In: A. Cremers, H. Kriegel (eds.) Theoretical Computer Science, *Lecture Notes in Computer Science*, vol. 145. Springer (1983), 6th GI-Conference, Dortmund, January 5–7, 1983
19. Kuçera, A., Slaman, T.A.: Randomness and recursive enumerability. SIAM J. Comput. **31**(1), 199–211 (2001)
20. Levin, L.: The concept of a random sequence. Dokl. Akad. Nauk SSSR **212**, 548–550 (1973)
21. Miller, J.S.: On work of Barmpalias and Lewis-Pye: A derivation on the d.c.e. reals. In: A. Day, M. Fellows, N. Greenberg, B. Khoussainov, A. Melnikov, F. Rosamond (eds.) Computability and Complexity: Essays Dedicated to Rodney G. Downey on the Occasion of His 60th Birthday, *Lecture Notes in Computer Science*, vol. 10010, pp. 644–659. Springer, Cham (2017)
22. Müller, N.T.: Subpolynomial complexity classes of real functions and real numbers. In: L. Kott (ed.) Proceedings of the 13th International Colloquium on Automata, Languages, and Programming, *Lecture Notes in Computer Science*, vol. 226, pp. 284–293. Springer, Berlin (1986)
23. Myhill, J.: Criteria of constructibility for real numbers. The Journal of Symbolic Logic **18**(1), 7–10 (1953)
24. Ng, K.M.: Some properties of d.c.e. reals and their degrees. Master's thesis, National University of Singapore (2005)
25. Pour-El, M.B., Richards, J.I.: Computability in Analysis and Physics. *Perspectives in Mathematical Logic*. Springer, Berlin (1989)
26. Rademacher, H.: On the Bloch-Landau constant. American Journal of Mathematics **65**, 387–390 (1943)
27. Raichev, A.: Relative randomness and real closed fields. Journal of Symbolic Logic **70**(1), 319–330 (2005)
28. Rettinger, R.: Bloch's constant is computable. Journal of Universal Computer Science **14**(6), 896–907 (2008)
29. Rettinger, R.: On computable approximations of Landau's constant. Log. Methods Comput. Sci. **8**(4), 4:15, 11 (2012)
30. Rettinger, R.: Effective Riemann mappings of multiply connected domains and Riemann surfaces. Mathematical Structures in Computer Science **27**(8), 1495–1520 (2017)
31. Rettinger, R., Zheng, X.: On the Turing degrees of divergence bounded computable reals. In: S.B. Cooper, B. Löwe, L. Torenvliet (eds.) New Computational Paradigms, *Lecture Notes in Computer Science*, vol. 3526, pp. 418–428. Springer, Berlin (2005), First Conference on Computability in Europe, CiE 2005, Amsterdam, The Netherlands, June 2005
32. Rettinger, R., Zheng, X.: Solovay reducibility on d-c.e. real numbers. In: L. Wang (ed.) Computing and Combinatorics, *Lecture Notes in Computer Science*, vol. 3595, pp. 359–368. Springer, Berlin (2005), 11th Annual International Conference, COCOON 2005, Kunming, China, August 16–19, 2005
33. Rettinger, R., Zheng, X.: A hierarchy of Turing degrees of divergence bounded computable real numbers. Journal of Complexity **22**(6), 818–826 (2006)

34. Rettinger, R., Zheng, X., Gengler, R., von Braunmühl, B.: Weakly computable real numbers and total computable real functions. In: J. Wang (ed.) Computing and Combinatorics, *Lecture Notes in Computer Science*, vol. 2108, pp. 586–595. Springer, Berlin (2001), 7th Annual International Conference, COCOON 2001, Guilin, China, August 20-23, 2001
35. Rice, H.G.: Recursive real numbers. Proc. Amer. Math. Soc. **5**, 784–791 (1954)
36. Robinson, R.: Review of "Peter, R., Rekursive Funktionen". The Journal of Symbolic Logic **16**, 280–282 (1951)
37. Soare, R.: Recursion theory and Dedekind cuts. Trans. Amer. Math. Soc. **140**, 271–294 (1969)
38. Soare, R.I.: Recursively Enumerable Sets and Degrees. *Perspectives in Mathematical Logic*. Springer, Berlin (1987)
39. Solovay, R.: Draft of a paper (or a series of papers) on Chaitin's work ... (1975), manuscript, IBM Thomas J. Watson Research Center, Yorktown Heights, NY, p. 215
40. Specker, E.: Nicht konstruktiv beweisbare Sätze der Analysis. The Journal of Symbolic Logic **14**(3), 145–158 (1949)
41. Turing, A.M.: On computable numbers, with an application to the "Entscheidungsproblem". Proceedings of the London Mathematical Society **42**(2), 230–265 (1937)
42. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)
43. Weihrauch, K., Zheng, X.: A finite hierarchy of the recursively enumerable real numbers. In: L. Brim, J. Gruska, J. Zlatuška (eds.) Mathematical Foundations of Computer Science 1998, *Lecture Notes in Computer Science*, vol. 1450, pp. 798–806. Springer, Berlin (1998), 23rd International Symposium, MFCS'98, Brno, Czech Republic, August, 1998
44. Zheng, X.: On the divergence bounded computable real numbers. In: T. Warnow, B. Zhu (eds.) Computing and Combinatorics, *Lecture Notes in Computer Science*, vol. 2697, pp. 102–111. Springer, Berlin (2003), COCOON 2003, 9th Annual International Conference, July 25-28, 2003, Big Sky, MT, USA
45. Zheng, X.: On the Turing degrees of weakly computable real numbers. Journal of Logic and Computation **13**(2), 159–172 (2003)
46. Zheng, X.: Classification of computably approximable real numbers. Theory of Computing Systems **43**(3-4), 603–624 (2008)
47. Zheng, X., Lub, D., Bao, K.: Divergence bounded computable real numbers. Theoretical Computer Science **351**(1), 27–38 (2006)
48. Zheng, X., Rettinger, R.: Weak computability and representation of reals. Mathematical Logic Quarterly **50**(4,5), 431–442 (2004)
49. Zheng, X., Rettinger, R.: On the extensions of Solovay reducibility. In: K.Y. Chwa, J.I. Munro (eds.) Computing and Combinatorics, *Lecture Notes in Computer Science*, vol. 3106, pp. 360–369. Springer, Berlin (2005), 10th Annual International Conference, COCOON 2004, Jeju Island, Korea, August 17–20, 2004
50. Zheng, X., Rettinger, R.: Convergence dominated reducibility and divergence bounded computable real numbers. (In preparation) (2017)
51. Zheng, X., Rettinger, R., Gengler, R.: Closure properties of real number classes under CBV functions. Theory of Computing Systems **38**, 701–729 (2005)

Chapter 2
Computability of Subsets of Metric Spaces

Zvonko Iljazović and Takayuki Kihara

Abstract We present a survey on computability of subsets of Euclidean space and, more generally, computability concepts on metric spaces and their subsets. In particular, we discuss computability of points in co-c.e. closed sets, representations of hyperspaces, Borel codes, computability of connectedness notions, classification of Polish spaces, computability of semicomputable sets, continua and manifolds, properties of computable images of a segment, and computability structures.

2.1 Introduction

To investigate computability in analysis and related areas, we need a language for talking about computability of complex numbers, compact sets, manifolds, etc. There is a general consensus regarding computability of real and complex numbers. However, what do we mean by a computable compact set, a computable measurable set, a computable Borel set, a computable manifold, and so on?

There have already been a number of reasonable answers to these questions. There are also various introductory materials on computability of basic concepts in analysis and related fields, cf. [82, 95, 12, 10, 78, 81]. This survey collects the answers to the above questions from a modern perspective.

Computable analysis has become a flourishing field; as a result research is very diverse. Each researcher needs a notion of computability at an appropriate level of abstraction. Therefore, in this survey, we introduce the notion of computability of

Zvonko Iljazović
Department of Mathematics, Faculty of Science, University of Zagreb, Croatia
e-mail: zilj@math.hr

Takayuki Kihara
Department of Mathematical Informatics, Graduate School of Informatics, Nagoya University, Japan
e-mail: kihara@i.nagoya-u.ac.jp

V. Brattka, P. Hertling (eds.), *Handbook of Computability and Complexity in Analysis*,
Theory and Applications of Computability, https://doi.org/10.1007/978-3-030-59234-9_2

sets step by step, namely, from subsets of Euclidean spaces to subsets of abstract represented spaces.

However, the real purpose of this survey is not merely to introduce fundamental notions of computability in analysis. In accordance with its rapid progress, computable analysis has been becoming a mature field. A large part of research is no longer at the stage of discussing basic definitions and producing expected results, but at the stage of producing unexpected, surprising results with sophisticated techniques.

From the authors' own point of view, we select remarkable results related to computability of sets (the selection is by no means exhaustive, of course), and attempt to sketch how the computability notions have brought us an enormous number of highly nontrivial and astonishing results.

Here we summarize the structure of this survey. In Section 2.2, we discuss the notion of computability of a subset of Euclidean space. Then, in Section 2.3, we introduce various notions of computability of subsets of computable metric spaces. In Section 2.4, we give a survey on degrees of noncomputability of points in co-c.e. closed sets (also known as Π^0_1 classes) from the perspective of computable analysis. In Section 2.5 we introduce the notion of computability of closed and compact sets in more abstract settings. Namely, we consider the hyperspaces of closed sets and compact sets, which enable us to introduce computability of sets as computability of points in hyperspaces. We also consider computability of Borel sets in Section 2.5.3. In Section 2.6, we consider computability of path-connectivity, local connectivity, etc. In Section 2.7, we sketch how the structure of degrees of noncomputability of points in a Polish space is affected by the global structure of the space itself with the emphasis on topological dimension theory. In Section 2.8, we give a survey on various conditions under which a semicomputable set is computable. In Section 2.9, we state some results about computable images of a segment. In Section 2.10, we consider computability structures on metric spaces.

2.2 Computable Subsets of Euclidean Space

In this section we discuss several natural ways to define the notion of a computable subset of Euclidean space.

A real number x is computable if it can be effectively approximated by a rational number with arbitrary precision. A point x in Euclidean space $\mathbb{R}^n$ is computable if it can be effectively approximated by a rational point $q \in \mathbb{Q}^n$ with arbitrary precision. Similarly, we may say that a subset S of $\mathbb{R}^n$ is computable if it can be effectively approximated by rational points with arbitrary precision. Of course, here we need to make precise what an effective approximation by rational points means.

Let d be the Euclidean metric on $\mathbb{R}^n$. Let $A, B \subseteq \mathbb{R}^n$ and $\varepsilon > 0$. We will say that A and B are ε-*close* if for each $x \in A$ there exists $y \in B$ such that $d(x,y) < \varepsilon$ and for each $y \in B$ there exists $x \in A$ such that $d(x,y) < \varepsilon$.

It is easy to conclude that for each compact set $S \subseteq \mathbb{R}^n$ and each $\varepsilon > 0$ there exists a finite subset A of $\mathbb{Q}^n$ such that S and A are ε-close. In view of this, it is natural to define that a compact set $S \subseteq \mathbb{R}^n$ is computable if for each $k \in \mathbb{N}$ we can effectively find a finite subset A_k of $\mathbb{Q}^n$ such that S and A_k are 2^{-k}-close. Intuitively, the finite set of points with rational coordinates A_k represents the image of the set S and this image becomes sharper as k becomes larger.

Another way to define the notion of a computable subset of Euclidean space is to follow the standard definition of a computable subset of $\mathbb{N}^n$. In classical computability theory a set $S \subseteq \mathbb{N}^n$ is computable if its characteristic function $\chi_S : \mathbb{N}^n \to \mathbb{N}$ is computable. However, if $S \subseteq \mathbb{R}^n$, $S \neq \emptyset$, $S \neq \mathbb{R}$, then the function $\chi_S : \mathbb{R}^n \to \mathbb{R}$ is not continuous and hence not computable. Therefore, it does not make sense to define that a subset of Euclidean space is computable if its characteristic function is computable. However, there is a suitable replacement for the characteristic function, namely for $S \subseteq \mathbb{R}^n$, $S \neq \emptyset$, we may consider the distance function $d_S : \mathbb{R}^n \to \mathbb{R}$ defined by

$$d_S(x) = d(x, S).$$

It is reasonable to consider here closed sets since they are uniquely determined by their distance functions. We will say that a closed set $S \subseteq \mathbb{R}^n$ is computable if the function $d_S : \mathbb{R}^n \to \mathbb{R}$ is computable. Intuitively, this means that for a given $x \in \mathbb{R}^n$ we can compute how close x lies to S (although, in general, we cannot effectively determine whether $x \in S$ or $x \notin S$).

To introduce the notion of a computable subset of $\mathbb{R}^n$ we may also proceed in the following way. We first define the notion of a computably enumerable (c.e.) subset of $\mathbb{R}^n$ and then we define that $S \subseteq \mathbb{R}^n$ is computable if S and $\mathbb{R}^n \setminus S$ are c.e. (following the classical fact: $S \subseteq \mathbb{N}^n$ is computable if S and $\mathbb{N}^n \setminus S$ are c.e.).

A subset S of $\mathbb{R}^n$ may be uncountable, so it does not make much sense to define that S is c.e. if it is the image of a computable function $\mathbb{N} \to \mathbb{R}^n$. What we can do here is to define that S is c.e. if it is the closure of the image of such a function (or $S = \emptyset$). In other words, S is c.e. if $S = \emptyset$ or there exists a computable sequence in $\mathbb{R}^n$ which is dense in S. It is also reasonable to assume that S is closed (in this case the given sequence uniquely determines S).

On the other hand, if S is closed, $\mathbb{R}^n \setminus S$ is open and to define that $\mathbb{R}^n \setminus S$ is c.e. we need another notion of computable enumerability. An open set $U \subseteq \mathbb{R}^n$ will be called c.e. open if it can be effectively exhausted by open balls. More precisely, U is c.e. open if $U = \emptyset$ or

$$U = \bigcup_{i \in \mathbb{N}} B(x_i, r_i),$$

where (x_i) is a computable sequence in $\mathbb{R}^n$ and (r_i) a computable sequence of positive real numbers. Here, for $a \in \mathbb{R}^n$ and $s > 0$, $B(a, s)$ denotes the open ball of radius s centered in a. So, we will say that $S \subseteq \mathbb{R}^n$ is computable if S is c.e. closed and $\mathbb{R}^n \setminus S$ is c.e. open.

The second and third definition given in this section coincide, and all three definitions coincide if S is compact [12]. In the next section we examine computability of sets in more general ambient spaces, namely in computable metric spaces.

2.3 Computable Metric Spaces

To describe various computability notions in Euclidean space, such as those given in the previous section, we actually only have to fix some effective enumeration $\alpha : \mathbb{N} \to \mathbb{Q}^n$ of $\mathbb{Q}^n$ or, more generally, some computable sequence $\alpha : \mathbb{N} \to \mathbb{R}^n$ whose image is dense in $\mathbb{R}^n$. This motivates the study of the notion of a computable metric space.

A *computable metric space* is a triple (X,d,α), where (X,d) is a metric space and $\alpha = (\alpha_i)$ is a sequence in X whose image is dense in (X,d) and such that the function $\mathbb{N}^2 \to \mathbb{R}$, $(i,j) \mapsto d(\alpha_i,\alpha_j)$, is computable. If d is a complete metric, then we also say that (X,d,α) is a *computable Polish space*. For an introduction to computable metric spaces, we refer the reader to [5, 94, 12, 71, 29].

Let (X,d,α) be a computable metric space. A point $x \in X$ is said to be *computable* in (X,d,α) if there exists a computable function $f : \mathbb{N} \to \mathbb{N}$ such that $d(x,\alpha_{f(k)}) < 2^{-k}$ for each $k \in \mathbb{N}$. A sequence (x_i) in X is said to be *computable* in (X,d,α) if there exists a computable function $f : \mathbb{N}^2 \to \mathbb{N}$ such that $d(x_i,\alpha_{f(i,k)}) < 2^{-k}$ for all $i,k \in \mathbb{N}$.

Example 2.3.1. Let $n \in \mathbb{N}$, $n \geq 1$, let $\alpha : \mathbb{N} \to \mathbb{Q}^n$ be a computable surjection and let d be the Euclidean metric on $\mathbb{R}^n$. Then $(\mathbb{R}^n,d,\alpha)$ is a computable metric space. It is easy to conclude that $x \in \mathbb{R}^n$, $x = (x_1,\ldots,x_n)$, is a computable point in $(\mathbb{R}^n,d,\alpha)$ if and only if $x_1,\ldots,x_n$ are computable numbers. Moreover, a sequence (x_i) in $\mathbb{R}^n$ is computable in $(\mathbb{R}^n,d,\alpha)$ if and only if the component sequences of (x_i) are computable as functions $\mathbb{N} \to \mathbb{R}$. We say that $(\mathbb{R}^n,d,\alpha)$ is the *computable Euclidean space*.

A *computable normed space* $(X,\|\cdot\|,e)$ is a separable normed space $(X,\|\cdot\|)$ together with a numbering $e : \mathbb{N} \to X$ such that the linear span of $\mathrm{rng}(e)$ is dense in X, and the induced metric space is a computable metric space. A complete computable normed space is called a *computable Banach space*. If a computable normed space is also a Hilbert space, then it is called a *computable Hilbert space*. For basics on these notions, see also Pour-El and Richards [82].

2.3.1 Computable Compact and Closed Sets

From now on, let (Δ_j) be some fixed effective enumeration of all finite subsets of $\mathbb{N}$. If (X,d,α) is a computable metric space and $j \in \mathbb{N}$, let

$$\Lambda_j = \{\alpha_i \mid i \in \Delta_j\}. \tag{2.1}$$

Clearly, (Λ_j) is an enumeration of all finite subsets of $\{\alpha_i \mid i \in \mathbb{N}\}$.

Let (X,d) be a metric space. That two subsets A and B of X are ε-*close* can be defined in the same way as in the case of Euclidean spaces (Section 2.2). For nonempty compact sets A and B in (X,d) we define their *Hausdorff distance*

$$d_H(A,B) = \inf\{\varepsilon > 0 \mid A \text{ and } B \text{ are } \varepsilon\text{-close}\}. \tag{2.2}$$

It is not hard to conclude that $d_H(A,B) < \varepsilon$ if and only if A and B are ε-close.

If (X,d,α) is a computable metric space and S a nonempty compact set in (X,d), then for each $\varepsilon > 0$ there exists $j \in \mathbb{N}$ such that $d_H(S,\Lambda_j) < \varepsilon$.

Let (X,d,α) be a computable metric space and let S be a compact set in (X,d). We say that S is a *computable compact set* in (X,d,α) if $S = \emptyset$ or there exists a computable function $f : \mathbb{N} \to \mathbb{N}$ such that $d_H(S,\Lambda_{f(k)}) < 2^{-k}$ for each $k \in \mathbb{N}$.

Example 2.3.2. Let (X,d,α) be a computable metric space and let $\mathscr{K}$ be the set of all nonempty compact sets in (X,d). Then the function $d_H : \mathscr{K} \times \mathscr{K} \to \mathbb{R}$ defined by (2.2) is a metric on $\mathscr{K}$ [74]. Let $\Lambda = (\Lambda_j)$ be the sequence defined by (2.1). It is easy to conclude that Λ is a dense sequence in the metric space $(\mathscr{K}, d_H)$. Furthermore, the function $\mathbb{N}^2 \to \mathbb{R}$, $(i,j) \mapsto d_H(\Lambda_i,\Lambda_j)$, is computable (see e.g. Proposition 2.5 in [41]). Hence $(\mathscr{K}, d_H, \Lambda)$ is a computable metric space. Note that computable points in $(\mathscr{K}, d_H, \Lambda)$ are exactly nonempty computable compact sets in (X,d,α).

A computable metric space (X,d,α) is said to be *effectively compact* if X is a computable compact set in (X,d,α).

If (X,d) is a metric space, $x \in X$ and $r > 0$, by $B(x,r)$ we will denote the open ball in (X,d) of radius r centered at x and by $\overline{B}(x,r)$ the corresponding closed ball.

Let (X,d,α) be a computable metric space, $n \in \mathbb{N}$ and $r \in \mathbb{Q}$, $r > 0$. We say that $B(\alpha_n, r)$ is a *rational open ball* in (X,d,α) and $\overline{B}(\alpha_n, r)$ a *rational closed ball* in (X,d,α). Let $\tau_1, \tau_2 : \mathbb{N} \to \mathbb{N}$ and $q : \mathbb{N} \to \mathbb{Q}$ be some fixed computable functions such that the image of q is the set of all positive rational numbers and $\{(\tau_1(i), \tau_2(i)) \mid i \in \mathbb{N}\} = \mathbb{N}^2$. For $i \in \mathbb{N}$ we define $\lambda_i = \alpha_{\tau_1(i)}$, $\rho_i = q_{\tau_2(i)}$ and

$$I_i = B(\lambda_i, \rho_i), \quad \hat{I}_i = \overline{B}(\lambda_i, \rho_i).$$

Then (I_i) is an enumeration of all rational open balls and $(\hat{I}_i)$ is an enumeration of all rational closed balls in (X,d,α).

Let (X,d,α) be a computable metric space and let S be a closed set in (X,d). We say that S is a *computably enumerable closed* set in (X,d,α) (or merely a *computably enumerable* set in (X,d,α)) if $\{i \in \mathbb{N} \mid I_i \cap S \neq \emptyset\}$ is a c.e. subset of $\mathbb{N}$.

Suppose (X,d,α) is a computable metric space, S a closed set in (X,d), and (x_j) a computable sequence in (X,d,α) which is dense in S, i.e. such that $S = \overline{\{x_j \mid j \in \mathbb{N}\}}$. (Here, by $\overline{A}$, for $A \subseteq X$, we denote the closure of A in the metric space (X,d).) Then S is a c.e. set in (X,d,α) [12]. So the following implication holds:

$$S \text{ contains a dense computable sequence } \Rightarrow S \text{ is c.e.} \tag{2.3}$$

The converse of implication (2.3) does not hold in general [4].

Let (X,d) be a metric space and $S \subseteq X$. We say that S is a *complete* set in (X,d) if $S = \emptyset$ or $S \neq \emptyset$ and $(S, d|_{S\times S})$ is a complete metric space.

If S is a complete set in a metric space (X,d), then S is closed in (X,d). Conversely, a closed set in (X,d) need not be complete, however if the metric space (X,d) is complete, then each closed set in (X,d) is complete.

Although the converse of the implication (2.3) does not hold in general, it does hold if S is a nonempty complete set. Hence, if (X,d,α) is a computable metric space and S is a nonempty c.e. set in this space which is complete in (X,d), then S contains a dense sequence which is computable in (X,d,α) (see [46]). In particular, if (X,d) is a complete metric space, then each nonempty c.e. set in (X,d,α) contains a dense computable sequence [12].

Let (X,d,α) be a computable metric space and let $U \subseteq X$. We say that U is a *computably enumerable open* set in (X,d,α) if there exists a c.e. set $A \subseteq \mathbb{N}$ such that $U = \bigcup_{i\in A} I_i$. We say that S is a *co-computably enumerable* (co-c.e.) *closed* set in (X,d,α) if $X \setminus S$ is a c.e. open set in (X,d,α).

Let (X,d,α) be a computable metric space and let $S \subseteq X$. We say that S is a *computable closed* set in (X,d,α) if S is c.e. and co-c.e. closed in (X,d,α).

Let (X,d,α) be a computable metric space, $n \geq 1$, and $B_1,\ldots,B_n$ rational open balls in this space. Then we say that $B_1 \cup \cdots \cup B_n$ is a *rational open set* in (X,d,α).

If (X,d,α) is a computable metric space and $j \in \mathbb{N}$, let

$$J_j = \bigcup_{i\in\Delta_j} I_i.$$

Then $\{J_j \mid j \in \mathbb{N}\}$ is the family of all rational open sets in (X,d,α).

Let (X,d,α) be a computable metric space and let K be a compact set in (X,d). We say that K is a *semicomputable compact set* in (X,d,α) if the set $\{j \in \mathbb{N} \mid K \subseteq J_j\}$ is c.e.

Less formally, K is semicomputable compact if we can effectively enumerate all rational open sets which cover K.

Let (X,d,α) be a computable metric space and let $K \subseteq X$. Then the following equivalence holds (see [41]):

$$K \text{ computable compact} \iff K \text{ c.e. and } K \text{ semicomputable compact.} \tag{2.4}$$

The notion of a semicomputable compact set can be generalized in the following way. Let (X,d,α) be a computable metric space and let $S \subseteq X$ be such that

(i) $S \cap B$ is a compact set in (X,d) for each closed ball B in (X,d);
(ii) the set $\{(i,j) \in \mathbb{N}^2 \mid S \cap \hat{I}_i \subseteq J_j\}$ is c.e.

Then we say that S is a *semicomputable set* in (X,d,α).

If S is compact in (X,d) and semicomputable in (X,d,α), then it is easy to conclude that S is semicomputable compact in (X,d,α). The converse of this implication also holds (see Proposition 3.3 in [15]), hence the following equivalence holds:

$$S \text{ compact and } S \text{ semicomputable} \iff S \text{ semicomputable compact.} \tag{2.5}$$

So, the notion of a semicomputable set generalizes the notion of a semicomputable compact set. In view of (2.4), we extend the notion of a computable compact set.

Let (X,d,α) be a computable metric space and let $S \subseteq X$. We say that S is a *computable set* in (X,d,α) if S is c.e. and semicomputable.

By (2.4) and (2.5) we have

$$S \text{ computable compact} \iff S \text{ compact and } S \text{ computable.}$$

Condition (i) from the definition of a semicomputable set easily implies that each semicomputable set in (X,d,α) is closed in (X,d). Moreover, we have the following result (see Proposition 3.5 in [15]).

Proposition 2.3.3. *Let (X,d,α) be a computable metric space. Then each semicomputable set in this space is co-c.e. closed. Consequently, each computable set in (X,d,α) is a computable closed set in (X,d,α).*

In general, a co-c.e. closed set need not be semicomputable. Also, a computable closed set need not be a computable set. Namely, in Example 3.2 in [40] a computable metric space $([0,b],d,\alpha)$ was constructed, where b is a positive real number and d is the Euclidean metric on $[0,b]$, such that $\{b\}$ is a co-c.e. closed set, but b is not a computable point in this space. In general, it is easy to conclude that in a computable metric space a point x is computable if and only if the set $\{x\}$ is semicomputable (see page 10 in [15]). Therefore, $\{b\}$ is not a semicomputable set in $([0,b],d,\alpha)$. Moreover, $[0,b]$ is a computable closed set in this space (in general, if (X,d,α) is a computable metric space, then X is clearly a computable closed set in (X,d,α)), but $[0,b]$ is not a computable set in this space: it is not semicomputable, which follows from Example 3.2 in [40].

However, under certain conditions on the ambient space, the notions of a semicomputable set and a co-c.e. closed set coincide.

Let (X,d,α) be a computable metric space such that the set $\{(i,j) \in \mathbb{N}^2 \mid \hat{I}_i \subseteq J_j\}$ is c.e. Then we say that (X,d,α) has the *effective covering property* [12].

The following theorem gives a sufficient condition that a computable metric space has the effective covering property (see [37]).

Theorem 2.3.4. *Let (X,d,α) be a computable metric space such that each closed ball in (X,d) is compact. Suppose that there exists a computable point a_0 and a computable sequence (x_i) in this space and a computable function $F : \mathbb{N}^2 \to \mathbb{N}$ such that $B(a_0,m) \subseteq \bigcup_{0\le i\le F(m,k)} B(x_i,2^{-k})$ for all $m,k \in \mathbb{N}$, $m \ge 1$. Then (X,d,α) has the effective covering property.*

Using Theorem 2.3.4, it is easy to conclude that the computable Euclidean space has the effective covering property.

Example 2.3.5. Let I^∞ denote the set of all sequences in $[0,1]$. It is known that the metric d on I^∞ defined by $d((x_i),(y_i)) = \sum_{i=0}^{\infty} \frac{1}{2^i}|x_i - y_i|$ induces a topology which coincides with the product topology on I^∞. The metric space (I^∞,d) is compact (Tychonoff's theorem) and it is called the *Hilbert cube*.

Let $r : \mathbb{N} \to \mathbb{Q}$ be a computable function whose range is $[0,1] \cap \mathbb{Q}$. Let $\sigma : \mathbb{N}^2 \to \mathbb{N}$ and $\eta : \mathbb{N} \to \mathbb{N}$ be computable functions such that each nonempty finite sequence in $\mathbb{N}$ equals $(\sigma(i,0),\dots,\sigma(i,\eta(i)))$ for some $i \in \mathbb{N}$ (such functions certainly exist). We define $\alpha : \mathbb{N} \to I^\infty$ by $\alpha_i = (r_{\sigma(i,0)},\dots,r_{\sigma(i,\eta(i))},0,0,\dots)$. Then (I^∞,d,α) is a computable metric space. Using Theorem 2.3.4 it is not hard to conclude that (I^∞,d,α) has the effective covering property (see [37]).

The proof of the next proposition can be found in [15] (Proposition 3.6).

Proposition 2.3.6. *Let (X,d,α) be a computable metric space which has the effective covering property and compact closed balls. Let $S \subseteq X$. Then S is co-c.e. closed if and only if S is semicomputable. Consequently, S is a computable closed set if and only if S is a computable set.*

2.4 Noncomputability of Points in Co-C.E. Closed Sets

2.4.1 Basis Theorems in Computability Theory

In classical computability theory, a lot of energy has been devoted to the study of the Turing degrees of points in subsets of an underlying space (mostly $2^{\mathbb{N}}$ or $\mathbb{N}^{\mathbb{N}}$). This field was pioneered by Kleene in the 1950s, who showed that

1. There is a nonempty co-c.e. closed subset of $\mathbb{R}$ with no computable points.
2. There is a nonempty co-c.e. closed subset of $\mathbb{N}^{\mathbb{N}}$ with no Δ^1_1 points.

The above results are sometimes referred to as Kleene's non-basis theorems. These theorems were a starting point of the long-running study of degrees of points in co-c.e. closed sets. As a second step, Kreisel proved the following basis theorems:

3. Every nonempty co-c.e. closed subset of $\mathbb{R}$ has a $\mathbf{0}'$-computable point.
4. Every nonempty co-c.e. singleton in $\mathbb{R}$ is computable.

These basis theorems fail for non-σ-compact spaces such as $\mathbb{N}^{\mathbb{N}}$. Indeed, Jockusch-McLaughlin [49] pointed out that for any computable ordinal α,

5. there is a co-c.e. singleton $\{x\}$ in $\mathbb{N}^{\mathbb{N}}$ such that x is not $\mathbf{0}^{(\alpha)}$-computable,

where $\mathbf{0}^{(\alpha)}$ is the α-th Turing jump. This kind of bad behavior of a co-c.e. closed set led us to the notion of semicomputability. For further studies on the degrees of co-c.e. singletons in $\mathbb{N}^{\mathbb{N}}$, see [19, 89] and [76, Chapters XII and XIII].

Regarding (3), the Kreisel basis theorem actually shows that the *leftmost* point of a nonempty co-c.e. closed set $P \subseteq [0,1]$ is *left-c.e.*, that is, the supremum of a computable sequence of rationals. It should be carefully noted that the notion "left-c.e." makes no sense at all in $[0,1]^n$ for $n \geq 2$.

In the higher-dimensional case, the following analog of left-c.e. is useful. For $n \leq \omega$, a point $x = (x_i)_{i<n} \in [0,1]^n$ is *n-left-CEA* if x_0 is left-c.e. and x_{i+1} is left-c.e. relative to x_i uniformly in i. More formally, there is a computable sequence $(g_i)_{i<n}$ of computable functions $g_i : [0,1]^i \to \mathbb{Q}^{\mathbb{N}}$ such that $x_i = \sup_n g_i(x_0,\dots,x_{i-1})(n)$.

Given $n \leq \omega$ and a nonempty closed set $P \subseteq [0,1]^n$, inductively define the *leftmost point* $(x_i)_{i<n}$ of P as follows. Define x_k as the smallest value such that P has a point whose first $k+1$ coordinates are $(x_0,\dots,x_k)$. By compactness of P, such a point exists. Then, it is easy to get a higher-dimensional analog of Kreisel's basis theorem.

Proposition 2.4.1 (see Kihara-Pauly [57]). *For any $n \leq \omega$, the leftmost point in a nonempty co-c.e. closed subset of $[0,1]^n$ is n-left-CEA.*

The Kreisel basis theorem has been refined by Jockusch-Soare's so-called low basis theorem. The importance of the low basis theorem is that the standard proof is applicable for any effectively compact computable metric space.

Given a computable metric space $\mathscr{X}$, its presentation automatically involves a computable list $(G_e)_{e\in\mathbb{N}}$ of c.e. open sets in $\mathscr{X}$. Then, the *Turing jump* of a point $x \in \mathscr{X}$ is defined by $x' = \{e \in \mathbb{N} : x \in G_e\}$. This generalization of the Turing jump has desirable properties; see Gregoriades-Kihara-Ng [28]. We say that a point $x \in \mathscr{X}$ is *low* if x' is Turing reducible to $\mathbf{0}'$.

Theorem 2.4.2 (Low Basis Theorem; Jockusch-Soare [50]). *Every nonempty co-c.e. closed set in an effectively compact computable metric space contains a low point.*

Proof. Let P be a nonempty co-c.e. closed subset of an effectively compact computable metric space $\mathscr{X}$. We construct a $\mathbf{0}'$-computable decreasing sequence $(Q_e)_{e\in\mathbb{N}}$ of co-c.e. closed sets in $\mathscr{X}$. Define $Q_0 = P$. By effective compactness, we can decide $Q_e \subseteq G_e$ using $\mathbf{0}'$ uniformly in e. Put $Q_{e+1} = Q_e$ if $Q_e \subseteq G_e$; otherwise, put $Q_{e+1} = Q_e \setminus G_e$. For any $z \in \bigcap_{e\in\mathbb{N}} Q_e$, clearly, $z'(e) = 1$ if and only if $Q_e \subseteq G_e$. We conclude that $z' \leq_T \mathbf{0}'$ since the latter condition is $\mathbf{0}'$-computable. □

A uniform version of the low basis theorem has also been proved by Brattka et al. [8]. As a historical remark, the original low basis theorem [50] was proved in the context of degrees of theories. A *PA-degree* is a Turing degree $\mathbf{d}$ such that every co-c.e. closed subset of $2^{\mathbb{N}}$ has a $\mathbf{d}$-computable point.

We shall emphasize that our introduction of basis theorems only scratches the surface of extremely deep studies on co-c.e. closed sets (also known as Π^0_1 classes). We refer the interested reader to Cenzer [16] and Diamondstone et al. [22] for more detailed introductions to the degree-theoretic analysis of co-c.e. closed sets. Detailed analysis of basis theorems has also been carried out from the perspective of *Medvedev degrees* and *Muchnik degrees*, cf. [88, 35, 34].

2.4.2 Basis Theorems in Computable Analysis

In computable analysis, we deal with a variety of geometric and topological properties of co-c.e. closed sets. When restricting our attention to co-c.e. closed sets possessing such global properties, basis and non-basis theorems often exhibit an interesting behavior. We first introduce a classical example in this direction. In the early stages of computability theory, global properties were always associated with *measure* and *category*. An example is as follows.

Theorem 2.4.3 (Kreisel-Lacombe [58]). *There is a co-c.e. closed subset of $[0,1]$ of positive Lebesgue measure that contains no computable point.*

A co-c.e. closed set as constructed in Theorem 2.4.3 is totally disconnected; otherwise, it contains a nonempty interval, and has a computable point. This trivial observation has became a source of new basis and non-basis theorems. From the geometric viewpoint, an interval is *convex*. From the topological viewpoint, an interval is *connected*. For the geometric side, Le Roux-Ziegler [61] observed that a nonempty convex co-c.e. closed set in $\mathbb{R}^n$ contains a computable point. It is possible to go further.

Theorem 2.4.4 (Neumann [75]). *Every nonempty convex co-c.e. closed subset of a finite-dimensional computable Banach space contains a computable point.*

Surprisingly, however, the same is not true in infinite-dimensional spaces. This fact is first implicitly mentioned by Miller [69]. Later it was shown that there is a computable dynamical system without computable invariant measures [25], where the set of invariant measures in a computable system forms a compact convex co-c.e. set.

Theorem 2.4.5 ([69, 25, 75]; see also Theorem 2.7.2). *There exists a nonempty convex co-c.e. closed subset of the Hilbert cube $[0,1]^{\mathbb{N}}$ containing no computable points.*

For the topological side, this naturally raises the question of whether every *connected* co-c.e. closed set contains a computable point. This is trivially false as pointed out by Le Roux-Ziegler [61].

Example 2.4.6. If A is a co-c.e. closed subset of $[0,1]$ with no computable element, then the *Cantor tartan* given by $([0,1]\times A)\cup(A\times[0,1])$ is a connected co-c.e. closed subset of $[0,1]^2$ with no computable points. Similarly, $([0,1]^2\times A)\cup([0,1]\times A\times[0,1])\cup(A\times[0,1]^2)$ is a simply connected co-c.e. closed subset of $[0,1]^3$ with no computable points.

A topological space X is *n-connected* if it is pathwise connected and $\pi_i(X)\equiv 0$ for any $1\leq i\leq n$, where $\pi_i(X)$ is the i-th homotopy group of X. A space X is *simply connected* if X is 1-connected. By a similar construction as in Example 2.4.6, one can get a nonempty n-connected, but not $(n+1)$-connected, co-c.e. closed set in $[0,1]^{n+2}$ which contains no computable points.

A space X is *contractible* if the identity map on X is null-homotopic. Note that, if X is contractible, then X is n-connected for each $n\geq 1$. A higher-dimensional variant of a Cantor tartan is never contractible. Thus, to construct a contractible co-c.e. closed set with no computable points, we need a different approach. By a *curve*, we mean a one-dimensional nondegenerate continuum. By a *continuum*, we mean a compact and connected metric space.

Theorem 2.4.7 (Kihara [53]). *There exists a contractible, co-c.e., planar curve which contains no computable points.*

Proof (Sketch). Let $C\subseteq[0,1]$ be a co-c.e. closed set with no computable points, and A be a computable arc, one of whose endpoints is non-computable (see Miller [68,

Example 4.1]). Imagine the cone space $(C \times A)/(C \times \{a\})$, where a is a unique non-computable endpoint of A. This gives us a Cantor fan with no computable points although it is unclear if it is co-c.e. or if it computably embeds into the Euclidean plane. However, a slight modification of this construction makes a fan co-c.e. in $[0,1]^2$. For more details, see [53]. □

As indicated in the above sketch, the example given by Kihara [53] is topologically homeomorphic to the Cantor fan. All known finite-dimensional co-c.e. closed sets with no computable points are not locally connected.

Question 2.4.8. Does there exist a nonempty, locally connected, co-c.e. closed subset of $[0,1]^n$ for some $n \in \mathbb{N}$ containing no computable points?

Note that Theorem 2.4.5 implies the existence of a nonempty, locally connected, co-c.e. closed subset of the Hilbert cube $[0,1]^{\mathbb{N}}$ which contains no computable points since every convex set is locally connected.

As a historical remark, basis theorems in classical computable analysis have sometimes been associated with mass problems. A *mass problem* is a subset of a (represented) space which appears as the set of solutions of a mathematical problem. Several mathematical problems in algebra, analysis, combinatorics, etc. have been found to be represented as co-c.e. closed subsets of certain computable metric spaces (cf. Cenzer-Remmel [17]). Degrees of difficulty of mass problems are often measured by Medvedev and Muchnik reducibility. We refer the reader to Simpson [86, 88] for Muchnik degrees of co-c.e. closed sets. The concept of mass problems is strongly associated with Reverse Mathematics [87], and the study of Weihrauch degrees as well (see the last Chapter in this handbook).

2.5 Represented Spaces and Uniform Computability

In previous sections, we only considered *non-uniform computability*. In this section, we introduce the notion of a represented space, which gives us a language for talking about *uniform computability*. For basics on represented spaces, see Weihrauch [95]. We also refer the reader to Pauly [78] for an excellent introduction to the theory of represented spaces.

Let $\mathscr{X} = (X,d,\alpha)$ be a computable metric space. Then, a *Cauchy name* of a point $x \in X$ is a sequence $p \in \mathbb{N}^{\mathbb{N}}$ such that $d(x,\alpha_{p(k)}) < 2^{-k}$ for any $k \in \mathbb{N}$. This notion induces a partial surjection $\delta :\subseteq \mathbb{N}^{\mathbb{N}} \to X$ defined by

$$\delta(p) = x \iff p \text{ is a Cauchy name of } x.$$

This surjection δ is called the *Cauchy representation of* X (induced from (d,α)).

In general, a *represented space* is a pair $\mathscr{X} = (X,\delta_X)$ of a set X and a partial surjection $\delta_X :\subseteq \mathbb{N}^{\mathbb{N}} \to X$. Such a δ_X is called a *representation of* X. If $\delta_X(p) = x$, then p is called a δ_X*-name of* x (or simply, a *name of* x if δ_X is clear from the context).

A point $x \in \mathscr{X}$ is *computable* if x has a computable name. A function $f : \mathscr{X} \to \mathscr{Y}$ is *computable* (*continuous*, resp.) if there is a partial computable (continuous, resp.) function on $\mathbb{N}^{\mathbb{N}}$ which, given a name of $x \in \mathscr{X}$, returns a name of $f(x) \in \mathscr{Y}$. In general, a partial function $\Phi :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ is a *realizer* of f if for any name p of $x \in \mathscr{X}$, $\Phi(p)$ is a name of $f(x)$. By definition, f is computable if and only if f has a computable realizer.

Remark 2.5.1. The notion of computability on computable metric spaces coincides with the notion of computability on represented spaces (w.r.t. the induced Cauchy representation). Thus, the theory of represented spaces generalizes the classical theory on computable metric spaces. Indeed, this epoch-making theory makes it possible to develop computability theory on an extremely wide class of topological spaces including various non-Hausdorff spaces, non-second-countable spaces, etc. cf. [78]. More precisely, Schröder [83, 84] showed that a T_0 space is admissibly represented if and only if it has a countable cs-network (a *cs-network* is a variant of Arhangel'skii's notion of a network introduced by Guthrie [31]).

Remark 2.5.2. As a related concept, the notion of a numbered set has been extensively studied in the theory of numberings (see Ershov [23]). A *numbered set* is a represented space (X, δ_X) such that the domain of δ_X is (effectively homeomorphic to) the natural numbers $\mathbb{N}$. There is a unification of these concepts. In realizability theory [77, 1], a represented space is called a *modest set*, and a multi-represented space is called an *assembly*. To be precise, a represented space is a modest set over Kleene's second (relative) algebra, i.e., the (relative) partial combinatory algebra given by Kleene's functional realizability, and a numbered set is a modest set over Kleene's first algebra, i.e., the pca given by Kleene's number realizability. We refer the reader to Bauer [1] for more details. This unification is useful since many generalized computation models such as infinite-time Turing machines induce pcas [2], and thus we do not have to reinvent the wheel for generalized computable analysis.

2.5.1 Represented Hyperspaces

The notion of a representation provides us with an abstract way of introducing computability on subsets of a space by considering a *represented hyperspace*.

By $A(X)$, we denote the set of all closed subsets of a computable metric space $\mathscr{X} = (X, d, \alpha)$. Recall that $(I_i)_{i\in\mathbb{N}}$ is a list of rational open balls in $\mathscr{X}$. For $p \in \mathbb{N}^{\mathbb{N}}$, we write $\mathrm{rng}(p) = \{p(n) - 1 : p(n) > 0\}$. We first introduce two representations ψ_+ and ψ_- of $A(X)$ capturing c.e. closed sets and co-c.e. closed sets, respectively.

$$\psi_+(p) = S \iff \mathrm{rng}(p) = \{n : S \cap I_n \neq \emptyset\},$$
$$\psi_-(p) = S \iff S = X \setminus \bigcup\{I_n : n \in \mathrm{rng}(p)\}.$$

The representations ψ_+ and ψ_- correspond to the lower Fell topology and the upper Fell topology on the hyperspace $A(X)$ of closed subsets of X (cf. [12]). We then

consider represented spaces $\mathscr{A}_+(\mathscr{X}) = (A(X), \psi_+)$, and $\mathscr{A}_-(\mathscr{X}) = (A(X), \psi_-)$. It is clear that the computable points in $\mathscr{A}_+(\mathscr{X})$ and $\mathscr{A}_-(\mathscr{X})$ are exactly the c.e. closed sets and the co-c.e. closed sets, respectively. One can also get a representation capturing computable closed sets as follows:

$$\psi_\pm(p \oplus q) = S \iff \psi_+(p) = \psi_-(q) = S,$$

where $(p \oplus q)(2n) = p(n)$ and $(p \oplus q)(2n+1) = q(n)$. Then, the computable points in $\mathscr{A}_\pm(\mathscr{X})$ are exactly the computable closed sets. Note that some authors use $\mathscr{A}(\mathscr{X})$ to denote $\mathscr{A}_\pm(\mathscr{X})$, while some other authors use $\mathscr{A}(\mathscr{X})$ to denote $\mathscr{A}_-(\mathscr{X})$.

We next introduce a representation of the hyperspace $K(X)$ of compact subsets of X. For a computable metric space $\mathscr{X} = (X, d, \alpha)$, recall that $(J_j)_{j\in\mathbb{N}}$ is the list of rational open sets in $\mathscr{X}$. Then, we define

$$\kappa_-(p) = S \iff \mathrm{rng}(p) = \{j \in \mathbb{N} : S \subseteq J_{p(j)}\}.$$

We also define $\kappa_\pm(p \oplus q) = S$ if and only if $\psi_+(p) = \kappa_-(q) = S$. We then define $\mathscr{K}_-(\mathscr{X}) = (K(X), \kappa_-)$ and $\mathscr{K}_\pm(\mathscr{X}) = (K(X), \kappa_\pm)$. The computable points in $\mathscr{K}_-(\mathscr{X})$ and $\mathscr{K}_\pm(\mathscr{X})$ are exactly the semicomputable compact sets and the computable compact sets, respectively.

The notion of a represented hyperspace enables us to discuss *uniform* computability of operations on closed and compact subsets of a computable metric space. For instance, consider the union and the intersection of co-c.e. closed sets. It is clear that if A and B are co-c.e. closed subsets of $\mathscr{X}$, so are $A \cup B$ and $A \cap B$. Indeed, the union and the intersection $\cup, \cap : \mathscr{A}_-(\mathscr{X}) \times \mathscr{A}_-(\mathscr{X}) \to \mathscr{A}_-(\mathscr{X})$ are computable, that is, given names of A and B, one can effectively find names of $A \cup B$ and $A \cap B$. In this way, the notion of computability on represented spaces automatically involves uniformity.

From the uniform perspective, the negative representation is quite well behaved. Actually, most basic operations on $\mathscr{A}_-(\mathscr{X})$ are known to be computable. For the positive representation, as shown in Brattka-Weihrauch [13], even the intersection $\cap : \mathscr{A}_\pm(\mathbb{R}^n)^2 \to \mathscr{A}_+(\mathbb{R}^n)$ is not computable. Indeed, for a T_1-space $\mathscr{X}$, $\cap : \mathscr{A}_+(\mathscr{X})^2 \to \mathscr{A}_+(\mathscr{X})$ is computable iff $\mathscr{X}$ is computably discrete [78].

There are a number of results regarding uniform computability of operations on hyperspaces. For instance, let chull be the map which, given a closed set, returns its *convex hull*. Then, $\mathrm{chull} : \mathscr{A}_+(\mathbb{R}^n) \to \mathscr{A}_+(\mathbb{R}^n)$ and $\mathrm{chull} : \mathscr{A}_-([0,1]^n) \to \mathscr{A}_-([0,1]^n)$ are computable [104, 60]. This useful result gives us a computable enumeration of all co-c.e. closed convex sets in $[0,1]^n$ while there is no limit computable way of deciding convexity of a co-c.e. closed set (cf. [81]). For further studies on computability of operations on hyperspaces, see Brattka-Presser [12]. For further reading on computability on other hyperspaces, we refer the reader to [32, 103, 104] for regular closed sets, and to [100, 101, 99] for measurable sets.

2.5.2 Represented Function Spaces

There is a way of viewing a hyperspace of closed sets as a function space. To see this, we first explain an important feature of represented spaces: *The category of represented spaces and (relatively) computable functions is cartesian closed*. This follows from the more general fact that the category $\mathrm{Mod}(\underset{\sim}{\Sigma},\Sigma)$ of modest sets over a relative pca $\langle\underset{\sim}{\Sigma},\Sigma\rangle$ is cartesian closed (cf. Bauer [1]). The following contains the details.

By Φ_e^z, we denote the e-th partial computable function on $\mathbb{N}^{\mathbb{N}}$ relative to an oracle $z\in\mathbb{N}^{\mathbb{N}}$. Let $e^\frown z$ denote the concatenation of e and z, that is, $(e^\frown z)(0)=e$ and $(e^\frown z)(n+1)=z(n)$. If $\mathscr{X}$ and $\mathscr{Y}$ are represented spaces, the set of relatively computable functions from $\mathscr{X}$ to $\mathscr{Y}$ is represented as follows:

$$\eta(e^\frown z)=f \iff \text{if } p \text{ is a name of } x\in\mathscr{X}, \text{ then } \Phi_e^z(p) \text{ is a name of } f(x)\in\mathscr{Y}.$$

In other words, Φ_e^z is a realizer of f. By $\mathscr{C}(\mathscr{X},\mathscr{Y})$ we denote the space of relatively computable functions from $\mathscr{X}$ to $\mathscr{Y}$ represented by η. Clearly, the computable points in $\mathscr{C}(\mathscr{X},\mathscr{Y})$ are exactly the computable functions from $\mathscr{X}$ to $\mathscr{Y}$.

Consider the set $S=\{\top,\bot\}$ represented by

$$\delta_S(p)=\begin{cases}\top & \text{if } (\exists n)\ p(n)\neq 0,\\ \bot & \text{if } (\forall n)\ p(n)=0.\end{cases}$$

We call $\mathbb{S}=(S,\delta_S)$ the *represented Sierpiński space* [83]. Assume that $\mathscr{X}$ is a represented space. Then, we can think of the function space $\mathscr{C}(\mathscr{X},\mathbb{S})$ as the *hyperspace of open sets in* $\mathscr{X}$, by identifying a function $f:\mathscr{X}\to\mathbb{S}$ with the open set $f^{-1}\{\top\}\subseteq\mathscr{X}$. Similarly, this space can also be viewed as the *hyperspace of closed sets*, by identifying a function $f:\mathscr{X}\to\mathbb{S}$ with the closed set $f^{-1}\{\bot\}\subseteq\mathscr{X}$. Via this identification, the representation η of the space $\mathscr{C}(\mathscr{X},\mathbb{S})$ yields the *Sierpiński representation* ψ_{Sier} of the hyperspace $A(X)$ of closed subsets of X as follows:

$$\psi_{\mathrm{Sier}}(p)=S \iff S=\eta(p)^{-1}\{\bot\}.$$

We now claim that ψ_{Sier} is equivalent to ψ_-. Given representations δ and η of a set X, we say that δ is *reducible to* η if there is a computable function which, given a δ-name of $x\in X$, returns an η-name of x. In other words, $\mathrm{id}:(X,\delta)\to(X,\eta)$ is computable. We say that δ is *equivalent to* η if δ is bireducible to η. Under this definition, one can show the following.

Proposition 2.5.3 (Brattka-Presser [12]). *The negative representation ψ_- of the hyperspace of closed subsets of a computable metric space is equivalent to the Sierpiński representation ψ_{Sier}.*

The represented hyperspace $\mathscr{K}_-(\mathscr{X})$ of compact sets can also be viewed as a function space. Let $\mathscr{O}(\mathscr{X})$ be the hyperspace of open sets in $\mathscr{X}$ represented as above, i.e., $\mathscr{O}(\mathscr{X})\simeq\mathscr{C}(\mathscr{X},\mathbb{S})$. A space $\mathscr{X}$ is compact iff the universal quantifier

$\forall_{\mathscr{X}} : \mathscr{O}(\mathscr{X}) \to \mathbb{S}$ is continuous, where $\forall_{\mathscr{X}}(X) = \top$ and $\forall_{\mathscr{X}}(U) = \top$ for $U \neq X$. Thus, a subset Y of $\mathscr{X}$ is compact iff $A_Y : \mathscr{O}(\mathscr{X}) \to \mathbb{S}$ is continuous, where

$$A_Y(U) = \begin{cases} \top & \text{if } Y \subseteq U, \\ \bot & \text{if } Y \not\subseteq U. \end{cases}$$

In other words, $Y \subseteq \mathscr{X}$ is compact iff $A_Y \in \mathscr{O}\mathscr{O}(\mathscr{X})$. Note that $A_Y = A_Z$ iff Y and Z have the same saturation (cf. [78]). Thus, this notion yields a representation $\kappa_\forall$ of saturated compact sets:

$$\kappa_\forall(p) = K \iff p \text{ is an } \mathscr{O}\mathscr{O}(\mathscr{X})\text{-name of } A_K.$$

One can easily see that $\kappa_\forall$ is equivalent to κ_- (for the hyperspace of compact subsets of a computable metric space).

The dual notion of compactness is known as overtness [92]. A space $\mathscr{X}$ is *overt*, iff the existential quantifier $\exists_{\mathscr{X}} : \mathscr{O}(\mathscr{X}) \to \mathbb{S}$ is continuous, where $\exists_{\mathscr{X}}(U) = \top$ for $U \neq \emptyset$ and $\exists_{\mathscr{X}}(\emptyset) = \bot$. Thus, a subset Y of $\mathscr{X}$ is overt iff $E_Y : \mathscr{O}(\mathscr{X}) \to \mathbb{S}$ is continuous, where

$$E_Y(U) = \begin{cases} \top & \text{if } Y \cap U \neq \emptyset, \\ \bot & \text{if } Y \cap U = \emptyset. \end{cases}$$

Although every subset $Y \subseteq \mathscr{X}$ is known to be overt, this definition yields a nontrivial (multi-)represented space $\mathscr{V}(\mathscr{X})$ of overt subsets of $\mathscr{X}$ by identifying $Y \subseteq \mathscr{X}$ with $E_Y \in \mathscr{O}\mathscr{O}(\mathscr{X})$. Obviously, $E_Y = E_Z$ iff Y and Z have the same topological closure; hence it induces a representation $\psi_\exists$ of the hyperspace of closed subsets of $\mathscr{X}$:

$$\psi_\exists(p) = Y \iff p \text{ is an } \mathscr{O}\mathscr{O}(\mathscr{X})\text{-name of } E_Y.$$

It is clear that $\psi_\exists$ is equivalent to the positive representation ψ_+ (for the hyperspace of closed subsets of a computable metric space).

In this way, computability theory on hyperspaces is absorbed into computability theory on function spaces. It should be carefully noted that the function-space representations ψ_{Sier}, $\kappa_\forall$, and $\psi_\exists$ are defined for the hyperspaces of *any* represented spaces, while the hyperspace representations ψ_-, ψ_+, κ_-, etc. make sense only for the hyperspaces of computable metric spaces. The representations introduced in this section are essentially due to Schröder [83]. The term "overt" is due to Taylor [92]. This framework has become fundamental in various contexts such as Escardó's *synthetic topology* [24] and Taylor's *abstract Stone duality* [92, 93]. See also Pauly [78] for a more detailed study of represented hyperspaces in the language of function spaces.

2.5.3 Borel Codes

A representation of Borel subsets of $\mathbb{R}$ (widely known as *Borel codes*) was first introduced by Solovay [90] to define the notion of a *random real over a model*. Solovay further explored the theory of Borel codes in his monumental work [91] on a model of Zermelo-Fraenkel (ZF) set theory in which all sets of reals are Lebesgue measurable. Since then, his representation of Borel sets has been a fundamental notion almost everywhere in set theory.

Here we only deal with Borel sets of finite rank. Let $\mathscr{X}$ be a computable metric space. We define representations σ^0_n and π^0_n of $\underset{\sim}{\Sigma}^0_n$ and $\underset{\sim}{\Pi}^0_n$ subsets of $\mathscr{X}$ as follows:

$$\pi^0_1(p) = \psi_-(p), \qquad \sigma^0_1(p) = \mathscr{X} \setminus \pi^0_1(p),$$
$$\pi^0_n(p) = \mathscr{X} \setminus \sigma^0_n(p), \qquad \sigma^0_{n+1}(p) = \bigcup_{i\in\mathbb{N}} \pi^0_n(p_i),$$

where recall that ψ_- is the negative representation of the hyperspace of closed subsets of $\mathscr{X}$. By $\underset{\sim}{\Sigma}^0_n(\mathscr{X})$ and $\underset{\sim}{\Pi}^0_n(\mathscr{X})$, we denote the hyperspaces of $\underset{\sim}{\Sigma}^0_n$ and $\underset{\sim}{\Pi}^0_n$ subsets of $\mathscr{X}$ represented by σ^0_n and π^0_n, respectively. By definition, $\underset{\sim}{\Pi}^0_1(\mathscr{X})$ is identical with $\mathscr{A}_-(\mathscr{X})$. In particular, the computable points in the spaces $\underset{\sim}{\Sigma}^0_1(\mathscr{X})$ and $\underset{\sim}{\Pi}^0_1(\mathscr{X})$ are the c.e. open sets and the co-c.e. closed sets, respectively. In general, a computable point in $\underset{\sim}{\Sigma}^0_n(\mathscr{X})$ ($\underset{\sim}{\Pi}^0_n(\mathscr{X})$, resp.) is called a Σ^0_n set (a Π^0_n set, resp.)

A function $f : \mathscr{X} \to \mathscr{Y}$ is $\underset{\sim}{\Sigma}^0_n$*-measurable* if the preimage of an open set under f is $\underset{\sim}{\Sigma}^0_n$. By second-countability of $\mathscr{Y}$, this is equivalent to saying that $f^{-1} : \underset{\sim}{\Sigma}^0_1(\mathscr{Y}) \to \underset{\sim}{\Sigma}^0_n(\mathscr{X})$ is continuous. We say that $f : \mathscr{X} \to \mathscr{Y}$ is Σ^0_n*-computable* if $f^{-1} : \underset{\sim}{\Sigma}^0_1(\mathscr{Y}) \to \underset{\sim}{\Sigma}^0_n(\mathscr{X})$ is computable (see Brattka [6]). Clearly, $\underset{\sim}{\Sigma}^0_1$-measurability and Σ^0_1-computability are equivalent to continuity and computability, respectively. The correspondence between the Borel hierarchy and the Baire hierarchy (the Banach-Hausdorff-Lebesgue theorem) was effectivized by Brattka as follows.

Theorem 2.5.4 (Brattka [6]). *Let $\mathscr{X}$ and $\mathscr{Y}$ be computable metric spaces, and let $k \geq 2$. Then, any Σ^0_{k+1}-computable function $f : \mathscr{X} \to \mathscr{Y}$ is the pointwise limit of a computable sequence of Σ^0_k-computable functions. For $\mathscr{X} = \mathbb{N}^{\mathbb{N}}$ this holds true in case $k = 1$ as well.*

In other words, Σ^0_{n+1}-computability is equivalent to n-th iterated limit computability (which can be thought of as a uniform version of the Shoenfield Limit Lemma). By using this notion, we can talk about the degrees of noncomputability of operations on represented spaces. Borel complexity of operations on represented hyperspaces has been studied by Gherardi [26] and Brattka-Gherardi [9].

Example 2.5.5 (Gherardi [26]). The intersection $\cap : \mathscr{A}(\mathbb{R}^n) \times \mathscr{A}(\mathbb{R}^n) \to \mathscr{A}(\mathbb{R}^n)$ is Σ^0_2-computable, but not computable. The intersection $\cap : \mathscr{A}_+(\mathbb{R}^n) \times \mathscr{A}_+(\mathbb{R}^n) \to \mathscr{A}_+(\mathbb{R}^n)$ is Σ^0_3-computable, but not Σ^0_2-computable.

In his work on functional analysis, Jayne [47] introduced a finer hierarchy of Borel functions. For $f : \mathscr{X} \to \mathscr{Y}$, we write $f^{-1}\underset{\sim}{\Sigma}^0_m \subseteq \underset{\sim}{\Sigma}^0_n$ if the preimage of a $\underset{\sim}{\Sigma}^0_m$

set under f is $\underset{\sim}{\Sigma}^0_n$. In this terminology, Σ^0_n-measurability is described as $f^{-1}\underset{\sim}{\Sigma}^0_1 \subseteq \underset{\sim}{\Sigma}^0_n$. By using Louveau's separation theorem [62] in effective descriptive set theory, Gregoriades-Kihara-Ng [28] showed that the property $f^{-1}\underset{\sim}{\Sigma}^0_m \subseteq \underset{\sim}{\Sigma}^0_n$ is equivalent to that $f^{-1} : \underset{\sim}{\Sigma}^0_m(\mathscr{Y}) \to \underset{\sim}{\Sigma}^0_n(\mathscr{X})$ has a Borel realizer. However, it is open whether the property $f^{-1}\underset{\sim}{\Sigma}^0_m \subseteq \underset{\sim}{\Sigma}^0_n$ is equivalent to that $f^{-1} : \underset{\sim}{\Sigma}^0_m(\mathscr{Y}) \to \underset{\sim}{\Sigma}^0_n(\mathscr{X})$ is continuous.

The Jayne-Rogers theorem [48] states that for a function f from an analytic subset $\mathscr{X}$ of a Polish space to a separable metric space, $f^{-1}\underset{\sim}{\Sigma}^0_2 \subseteq \underset{\sim}{\Sigma}^0_2$ if and only if it is closed-piecewise continuous, that is, there is a closed cover $(P_n)_{n\in\mathbb{N}}$ of $\mathscr{X}$ such that $f \restriction P_n$ is continuous for any $n \in \mathbb{N}$.

We consider effective versions of Jayne's Borel hierarchy and piecewise continuity. A *computable Π^0_n cover* of $\mathscr{X}$ is a computable sequence $(P_n)_{n\in\mathbb{N}}$ of Π^0_n subsets of $\mathscr{X}$ such that $\mathscr{X} = \bigcup_n P_n$. We say that $f : \mathscr{X} \to \mathscr{Y}$ is *Π^0_n-piecewise Σ^0_m-computable* if there is a computable Π^0_n cover of $\mathscr{X}$ such that the restriction $f \restriction P_n$ is Σ^0_m-computable uniformly in $n \in \mathbb{N}$. If $m = 1$, we simply say that f is Π^0_n-piecewise computable. The notion of Π^0_1-piecewise computability is equivalent to *computability with finite mindchanges*, which has turned out to be a very important notion in computable analysis (cf. [8, 14]). Then, the Jayne-Rogers theorem is effectivized as follows.

Theorem 2.5.6 (Pauly-de Brecht [79]). *Let $f : \mathscr{X} \to \mathscr{Y}$ be a function between computable metric spaces $\mathscr{X}$ and $\mathscr{Y}$. Then, $f^{-1} : \underset{\sim}{\Sigma}^0_2(\mathscr{Y}) \to \underset{\sim}{\Sigma}^0_2(\mathscr{X})$ is computable if and only if f is Π^0_1-piecewise computable.*

Soon after, Kihara [54] found that Theorem 2.5.6 can be generalized to higher Borel ranks whenever $\mathscr{X}$ and $\mathscr{Y}$ are finite-dimensional. The notion of topological dimension suddenly appeared out of nowhere! The reason was later clarified by Kihara-Pauly [57]: In [54], the Shore-Slaman join theorem in Turing degree theory was a key tool for generalizing Theorem 2.5.6; however, the degree structure of an infinite-dimensional computable metric space is generally different from the Turing degrees (see Sections 2.7.2 and 2.7.3). After this discovery, Gregoriades-Kihara-Ng [28] introduced a variant of Kumabe-Slaman forcing to generalize the Shore-Slaman join theorem in the setting of "infinite-dimensional" Turing degree theory, and then succeeded in removing the dimension-theoretic restriction from the former result [54].

Theorem 2.5.7 (Gregoriades-Kihara-Ng [28]). *Let $f : \mathscr{X} \to \mathscr{Y}$ be a function between computable Polish spaces $\mathscr{X}$ and $\mathscr{Y}$, and assume that $n < 2m$. Then, $f^{-1} : \underset{\sim}{\Sigma}^0_{m+1}(\mathscr{Y}) \to \underset{\sim}{\Sigma}^0_{n+1}(\mathscr{X})$ is computable if and only if f is Π^0_n-piecewise Σ^0_{n-m+1}-computable.*

An open problem is whether we can remove the assumption $n < 2m$ from Theorem 2.5.7. For other directions of investigation on piecewise computability, it is also important to think about decomposition into finitely many computable functions. For $k \in \mathbb{N}$, we say that $f : \mathscr{X} \to \mathscr{Y}$ is *k-Γ-piecewise computable* if there is an increasing sequence $(G_i)_{i<k}$ of Γ sets such that $\mathscr{X} = \bigcup_{i<k} G_i$ and $f \restriction G_i \setminus G_{i-1}$ is

computable. Then, $(k+1)$-Π^0_1-piecewise computability corresponds to *computability with at most k mindchanges* [14], and $(k+1)$-Δ^0_2-piecewise computability corresponds to *computability with finite mindchanges and at most k errors* [33]. For further reading on piecewise computability, see also de Brecht [14] and Kihara [55].

The notion of Borel codes in the context of represented spaces has also been studied in Gregoriades et al. [29] in detail. Borel codes are also used to introduce (multi-)representations of Borel-generated σ-ideals such as Lebesgue null sets and meager sets (cf. [56]). Representations of such σ-ideals are evidently useful when talking about randomness, genericity, forcing, etc. as Solovay did. This way of thinking has become ubiquitous in modern set theory.

For further direction, Pauly-de Brecht [80] recently proposed *synthetic descriptive set theory* as a reinterpretation of descriptive set theory (DST) in the category-theoretic context. One of the core ideas of synthetic DST is the use of endofunctors. An endofunctor is a functor from a category to itself. They introduced specific endofunctors relevant for the study of DST, e.g. the finite-mindchange endofunctor $^\nabla$ and the jump endofunctor $'$. Applying $^\nabla$ and $'$ to the hyperspace of open sets yields the hyperspaces of $\underset{\sim}{\Delta}^0_2$ sets and $\underset{\sim}{\Sigma}^0_2$ sets in the Borel hierarchy, respectively. In this way, synthetic DST provides a language for talking about descriptive set-theoretic concepts in a unified category-theoretic manner.

2.6 Computability of Connectedness Notions

The notion of a represented space is useful for introducing effective versions of various topological concepts. In this section, we will use represented spaces to introduce the notions of effective pathwise connectivity, effective local connectivity, etc., and then we will address a few computability-theoretic works involving these notions.

2.6.1 Effective Connectivity Properties

Consider the notion of pathwise connectivity and arcwise connectivity. Every arcwise connected space is pathwise connected, and the converse also holds for Hausdorff spaces. However, Miller [68] first observed that the notions of effective pathwise connectivity and effective arcwise connectivity do not coincide even for computable closed subsets of $[0,1]^2$.

Example 2.6.1 (Miller [68, Example 5.1]). There is a planar arc A with computable endpoints such that A is computable closed, A is the image of a computable function $f : [0,1] \to [0,1]^2$, but A cannot be the image of a computable injection $g : [0,1] \to [0,1]^2$.

Informally, we say that a space $\mathscr{X}$ is computably pathwise connected if, given (names of) points $x, y \in \mathscr{X}$, one can effectively find (a name of) a continuous function $f : [0,1] \to \mathscr{X}$ such that $f(0) = x$ and $f(1) = y$.

This yields the notion of computability of a *multi-valued* function. For represented spaces $\mathscr{X}$ and $\mathscr{Y}$, a multi-valued function $F : \mathscr{X} \rightrightarrows \mathscr{Y}$ is computable if there is a computable (single-valued) function Φ which, given a name of a point $x \in \mathscr{X}$, returns a name of an element of $F(x) \in \mathscr{Y}$. Note that Φ does not necessarily induce a function from $\mathscr{X}$ to $\mathscr{Y}$, that is, even if p_0 and p_1 are names of the same point $x \in \mathscr{X}$, $\Phi(p_0)$ and $\Phi(p_1)$ can be names of different points $y_0 \neq y_1$ in $F(x)$.

Brattka [7] formalized the notion of effective pathwise connectivity as follows. We say that a computable metric space $\mathscr{X}$ is *effectively pathwise connected* if the multi-valued function $F : \mathscr{X}^2 \rightrightarrows \mathscr{C}([0,1], \mathscr{X})$ defined by

$$F(x,y) = \{f \in \mathscr{C}([0,1], \mathscr{X}) : f(0) = x \text{ and } f(1) = y\}$$

is computable. Note that if F has a single-valued continuous selection, then $\mathscr{X}$ has to be contractible. Thus, multi-valuedness of the above definition is essential!

One can also define various different effectivizations of path/arcwise connectivity. For instance, a computable metric space $\mathscr{X}$ is *[co-c.e. arc-] connected* if the multi-valued function $F : \mathscr{X}^2 \rightrightarrows \mathscr{A}_-(\mathscr{X})$ defined by

$$F(x,y) = \{S \in \mathscr{A}_-(\mathscr{X}) : x, y \in S \text{ and } S \text{ is an arc}\}$$

is computable. It is easy to see that there is a planar curve A such that A is computable closed, effective pathwise connected, but not [co-c.e. arc-] connected [53].

2.6.2 Computable Graph Theorem

In classical computability theory a function $f : \mathbb{N} \to \mathbb{N}$ is computable if and only if its graph is computable. On the other hand, if X and Y are topological spaces such that Y is compact and Hausdorff, then a function $f : X \to Y$ is continuous if and only if its graph is a closed set in $X \times Y$ [73]. The question is what can be said about a computable version of this result, i.e. if X and Y are computable metric spaces and $f : X \to Y$, under what assumptions the computability of f is equivalent to the fact that the graph of f is a computable closed set.

If (X, d, α) and (Y, d', α') are computable metric spaces, we define their *product* as the computable metric space $(X \times Y, d'', \alpha'')$ defined by

$$d''((x,y),(x',y')) = \max\{d(x,x'), d'(y,y')\} \text{ and } \alpha''\langle i,j\rangle = (\alpha(i), \alpha'(j)).$$

Let (X, d, α) be a computable metric space and let $\mathscr{O}$ be the family of all open sets in (X, d). Let δ be the representation of $\mathscr{O}$ defined by

$$\delta(p) = \bigcup \{I_n \mid n \in \mathrm{rng}(p)\}.$$

We say that (X,d,α) is *effectively locally connected* [7] if there exists a computable multivalued function $C :\subseteq X \times \mathbb{R} \rightrightarrows \mathscr{O}$ such that for all $x \in X$ and $r > 0$ the set $C(x,r)$ is nonempty and each $U \in C(x,r)$ is a connected set such that $x \in U \subseteq B(x,r)$.

Theorem 2.6.2 (Brattka [7]). *Let X and Y be computable metric spaces. Then the function*

$$\text{graph} : \mathscr{C}(X,Y) \to \mathscr{A}_{\pm}(X \times Y), \quad f \mapsto \text{graph}(f) \tag{2.6}$$

is computable.

(i) If Y is effectively compact, then the partial inverse $\text{graph}^{-1} :\subseteq \mathscr{A}_{-}(X,Y) \to \mathscr{C}(X,Y)$ *of the map (2.6) is computable.*

(ii) If X is effectively locally connected and $Y = \mathbb{R}^n$ for some $n \geq 1$, then the partial inverse $\text{graph}^{-1} :\subseteq \mathscr{A}_{\pm}(X,Y) \to \mathscr{C}(X,Y)$ *of the map (2.6) is computable.*

(iii) If X is effectively pathwise connected and $n \geq 1$, then the map

$$F :\subseteq \mathscr{A}_{-}(X \times \mathbb{R}^n) \times X \times \mathbb{R}^n \to \mathscr{C}(X,\mathbb{R}^n), \quad (A,a,b) \mapsto \text{graph}^{-1}(A),$$

which is defined for all (A,a,b) such that $A = \text{graph}(f)$ and $f(a) = b$ for some $f \in \mathscr{C}(X,\mathbb{R}^n)$, is computable.

As a consequence of Theorem 2.6.2 we get the following result.

Corollary 2.6.3 (Brattka [7]). *Let X and Y be computable metric spaces. Suppose Y is effectively compact and $f : X \to Y$. Then f is computable if and only if* $\text{graph}(f)$ *is co-c.e. closed and if and only if* $\text{graph}(f)$ *is computable closed. We get the same conclusion if we assume that X is effectively pathwise connected and $Y = \mathbb{R}^n$ for some $n \geq 1$.*

The additional assumptions on the computable metric spaces in the statement of Corollary 2.6.3 cannot be omitted: in general it is possible that f is not computable although it is continuous and $\text{graph}(f)$ is computable closed [7].

A computable metric space (X,d,α) is said to be *locally computable* if for each compact set A in (X,d) there exists a computable compact set K in (X,d,α) such that $A \subseteq K$. In the following theorem we get the same conclusion as in Corollary 2.6.3 but with different assumptions.

Theorem 2.6.4 (Brattka [7]). *Let X and Y be computable metric spaces such that X is compact and Y is locally computable. Let $f : X \to Y$ be a continuous function. Then f is computable if and only if* $\text{graph}(f)$ *is co-c.e. closed and if and only if* $\text{graph}(f)$ *is computable closed.*

2.6.3 Degrees of Difficulty

A new paradigm brought from the theory of representations enables us to talk about the degrees of difficulty of problems involving hyperspaces. For instance,

1. Find a connected component of a given nonempty co-c.e. closed subset of $[0,1]^n$.
2. Find a nontrivial subcontinuum of a given co-c.e. closed subset of $[0,1]^{\mathbb{N}}$ of positive dimension.

The problem (1) has been studied by Le Roux-Ziegler [61] and Brattka et al. [11]. If a compact metric space is not zero-dimensional, it always has a nondegenerate subcontinuum. The problem (2) has been studied by Kihara [52].

In general topology, there are various strengthenings of connectivity. One of those is the notion of a Cantor manifold, which was introduced by Urysohn as one of the most fundamental notions in topological dimension theory. We say that a topological space $\mathscr{X}$ is disconnected by $A \subseteq \mathscr{X}$ if $\mathscr{X} \setminus A$ is a union of disjoint open sets. It is clear that a space is disconnected iff it is disconnected by the empty set $\emptyset$. An *n-dimensional Cantor manifold* is an n-dimensional compact space which is not disconnected by an at most $(n-2)$-dimensional subset.

Kihara [52] recently noticed that the notion of a Cantor manifold has an application in the study of degrees of points in computable metric spaces. A key tool is the Hurewicz-Tumarkin Cantor manifold theorem, which says that every n-dimensional compact metric space contains an n-dimensional Cantor manifold (cf. [36, 67]). An interesting open question is to determine the degree of difficulty of the Cantor manifold theorem, for instance as follows.

Question 2.6.5. Let $P \subseteq [0,1]^{\mathbb{N}}$ be a co-c.e. closed set of positive dimension. Does every PA-degree compute an $\mathscr{A}_{-}$-name of a Cantor submanifold of P?

2.7 Classification of Polish Spaces

2.7.1 Borel Isomorphism Theorem

Kuratowski's Borel isomorphism theorem is one of the most fundamental theorems on Polish spaces; it says that every uncountable Polish space is Borel isomorphic to $\mathbb{R}$. For an effective counterpart, if we replace "uncountable" with "perfect," it is known that every perfect computable Polish space is Δ^1_1-isomorphic to $\mathbb{R}$ (see Moschovakis [71, Section 3I]). However, Gregoriades [27] showed that perfectness is essential for effectivity.

Theorem 2.7.1 (Gregoriades [27]). *There exists a zero-dimensional, uncountable, computable Polish space which is not Δ^1_1-isomorphic to $\mathbb{R}$.*

Proof. Let $T \subseteq \omega^{<\omega}$ be Kleene's tree, none of whose infinite paths is Δ^1_1, and let $[T]$ be the set of infinite paths through T. Consider $\mathscr{T} = T \cup [T]$, where a basic open set is the set of all extensions of a finite string in T or the singleton consisting of a finite string. It is easy to give a computable Polish metrization of $\mathscr{T}$. Then T is a c.e. open set consisting of isolated points in $\mathscr{T}$. By our choice of T, the Δ^1_1 points in $\mathscr{T}$ are exactly T; hence, c.e. open in $\mathscr{T}$. Therefore, $\mathscr{T}$ is not Δ^1_1-isomorphic to $\mathbb{R}$ since the set of all Δ^1_1 points in $\mathbb{R}$ is not Δ^1_1. □

Gregoriades [27] then studied the Δ^1_1-embeddability order on (zero-dimensional) computable Polish spaces, and showed that every countable partial order can be embedded into the order. The proof requires a detailed analysis of (hyper)degrees of points in computable Polish spaces. All spaces that appear in his proof computably embed into $\mathbb{N}^{\mathbb{N}}$, and thus, one can just adopt classical degree theory. However, as we will see later, exploring degree theory in an arbitrary computable metric space leads us to the discovery of a new connection between computability and dimension.

2.7.2 Continuous Degree Theory

Recall that every point in a computable metric space is named by elements in $\mathbb{N}^{\mathbb{N}}$ via the Cauchy representation. We estimate how complicated a point in a computable metric space is by considering the *degree of difficulty of calling a name* of the point. Of course, it is possible for each point to have many names, and this nature yields the phenomenon that there is a point with no easiest names with respect to Turing reducibility.

Let $\mathscr{X}$ and $\mathscr{Y}$ be computable metric spaces. A point $y \in \mathscr{Y}$ *is point-Turing reducible to* $x \in \mathscr{X}$ if there is a computable function Φ that, given a name p of x, returns a name $\Phi(p)$ of y. In other words, there is a partial computable function $f :\subseteq \mathscr{X} \to \mathscr{Y}$ such that $f(x) = y$. This notion was introduced by Miller [69] under the name of *representation reducibility* and *continuous degrees*. We say that a point $x \in \mathscr{X}$ *has a* $\mathscr{Y}$*-degree* if x is point-Turing equivalent to a point in $\mathscr{Y}$. Pour-El and Richards [82] observed that any point in Euclidean space has a $2^{\mathbb{N}}$-degree.

Miller's discovery of non-$2^{\mathbb{N}}$-degrees and the connection between such degrees and Scott ideals is an astounding achievement of the 2000s, which has brought a new paradigm in computability theory.

A *Turing ideal* is a set $\mathscr{I} \subseteq 2^{\mathbb{N}}$ which forms an ideal w.r.t. Turing reducibility $\leq_T$, that is, $x \leq_T y \in \mathscr{I}$ implies $x \in \mathscr{I}$, and $x, y \in \mathscr{I}$ implies $x \oplus y \in \mathscr{I}$. A *Scott ideal* is a Turing ideal $\mathscr{I}$ such that for any $x \in \mathscr{I}$, if $P \subseteq 2^{\mathbb{N}}$ is nonempty and co-c.e. closed relative to x (that is, P is computable relative to x as a point in $\mathscr{A}_-(2^{\mathbb{N}})$), then $P \cap \mathscr{I}$ is nonempty.

Theorem 2.7.2 (Miller [69]).

1. *The Hilbert cube* $[0,1]^{\mathbb{N}}$ *has a point of non-*$2^{\mathbb{N}}$*-degree.*
2. *If* $x \in [0,1]^{\mathbb{N}}$ *has a non-*$2^{\mathbb{N}}$*-degree, then the set of all* $y \in 2^{\mathbb{N}}$ *that are point-Turing reducible to* x *forms a Scott ideal.*
3. *For every countable Scott ideal* $\mathscr{I}$*, there is a point* $x \in [0,1]^{\mathbb{N}}$ *such that* $y \in \mathscr{I}$ *if and only if* y *is point-Turing reducible to* x*.*

Proof (for 1). Let Φ_e be the e-th partial computable function from $[0,1]^{\mathbb{N}}$ to $[0,1]$. By approximating partial computations, one can easily construct a computable function $\psi : [0,1]^{\mathbb{N}} \times \mathbb{N} \to \mathscr{A}_-([0,1])$ such that $\psi(x,e) = \{\Phi_e(x)\}$ if $x \in \mathrm{dom}(\Phi_e)$; otherwise, $\psi(x,e)$ is a nondegenerate interval. Then, $\Psi : [0,1]^{\mathbb{N}} \to \mathscr{A}_-([0,1]^{\mathbb{N}})$ defined

by $\Psi(x) = \prod_e \psi(x,e)$ is a computable function with a co-c.e. closed graph such that $\Psi(x)$ is nonempty and convex for any $x \in [0,1]^{\mathbb{N}}$. By Kakutani's fixed-point theorem, Ψ has a fixed-point $x \in \Psi(x)$. Note that there is a computable multi-valued function $F : [0,1]^{\mathbb{N}} \rightrightarrows [0,1]$ such that $F(x) \neq x(e)$ for any $e \in \mathbb{N}$. In particular, any name of x computes some $y \in [0,1]$ such that $y \notin \{x(e) : e \in \mathbb{N}\}$. If x has a $2^{\mathbb{N}}$-degree, x computes its name, and thus, computes such a y; however, by the property $x \in \Psi(x)$, if x computes y, then $y = x(e)$ for some e, a contradiction. □

Later, Day-Miller [21] observed a similar phenomenon in the theory of algorithmic randomness. The space of Borel probability measures on $2^{\mathbb{N}}$ is computably metrizable (e.g. via the Prokhorov metric). A probability measure μ is called *neutral* if every infinite binary sequence is Martin-Löf random w.r.t. μ. Day-Miller [21] noticed that a neutral measure cannot have a $2^{\mathbb{N}}$-degree, and hence, its lower Turing cone forms a Scott ideal as in Theorem 2.7.2 (2). They also showed an analog of Theorem 2.7.2 (3): For every Scott ideal $\mathscr{I}$, there is a neutral measure μ such that $y \in \mathscr{I}$ iff y is point-Turing reducible to μ.

2.7.3 Computable Aspects of Infinite Dimensionality

The previous works on continuous degrees [69, 21] make crucial use of the Kakutani fixed-point theorem (for infinite-dimensional spaces). This leads us to the conjecture that the notion of topological dimension is essential in degree theory on computable metric spaces. It is made more and more plausible by the recent works [54, 28] extending Theorem 2.5.6 and also by the following result.

Theorem 2.7.3 (Kihara-Pauly [57]). *The following are equivalent for computable metric spaces $\mathscr{X}$ and $\mathscr{Y}$:*

1. *Every $\mathscr{X}$-degree is a $\mathscr{Y}$-degree.*
2. *There is a countable partition $(\mathscr{X}_i)_{i\in\mathbb{N}}$ of $\mathscr{X}$ such that each $\mathscr{X}_i$ computably embeds into $\mathscr{Y}$.*

By Theorem 2.7.3 with $\mathscr{Y} = 2^{\mathbb{N}}$, we can characterize the Turing degrees in terms of topological dimension theory. A topological space is called *countable-dimensional* if it is a countable union of finite-dimensional subspaces. If $\mathscr{X}$ is Polish, this is equivalent to having transfinite small inductive dimension in the sense of Menger-Urysohn (cf. [36]). Then, relative to some oracle, all points in $\mathscr{X}$ have $2^{\mathbb{N}}$-degrees iff $\mathscr{X}$ is countable-dimensional. In particular, Theorem 2.7.2 (1) is a corollary of Theorem 2.7.3 since it is known that the Hilbert cube is not countable-dimensional (cf. [36, 67]). This simple observation completely solves a mystery about the occurrence of non-$2^{\mathbb{N}}$-degrees in the Hilbert cube (and the space of probability measures).

There is more circumstantial evidence that topological dimension theory is crucial to making a deep study of computable metric spaces. For $n \leq \omega$, an *n-left-CEA*

operator is a function $\Gamma : \mathbb{N}^{\mathbb{N}} \to [0,1]^n$ such that $\Gamma(x)$ is n-left-CEA uniformly relative to x. A *universal n-left-CEA operator* is an n-left-CEA operator $\Gamma : \mathbb{N}^{\mathbb{N}} \to [0,1]^n$ such that for any n-left-CEA operator Λ, there is an e such that $\Gamma(e^\frown x) = \Lambda(x)$ for any x. Kihara-Pauly [57] showed that the graph of a universal n-left-CEA operator (as a subspace of the Hilbert cube) has an interesting dimension-theoretic property.

1. The graph $\mathbb{G}_n$ of a universal n-left-CEA operator is a totally disconnected n-dimensional Polish space whose countable product $\mathbb{G}_n^{\mathbb{N}}$ is also n-dimensional.
2. The graph $\mathbb{G}_\omega$ of a universal ω-left-CEA operator is a totally disconnected infinite-dimensional Polish space.

We say that $\mathscr{X}$ is finite-level Borel isomorphic to $\mathscr{Y}$ if there is a bijection $f : \mathscr{X} \to \mathscr{Y}$ such that $f^{-1}\underset{\sim}{\Sigma}^0_n \subseteq \underset{\sim}{\Sigma}^0_n$ and $f\underset{\sim}{\Sigma}^0_n \subseteq \underset{\sim}{\Sigma}^0_n$ for some $n \in \mathbb{N}$. We also say that $\mathscr{X}$ finite-level Borel embeds into $\mathscr{Y}$ if $\mathscr{X}$ is finite-level Borel isomorphic to a Borel subset of $\mathscr{Y}$ of finite rank. By applying Theorems 2.5.7, 2.7.2, and 2.7.3, one can show that the graph of a universal ω-left-CEA operator has an intermediate finite-level Borel isomorphism type.

Theorem 2.7.4 (Kihara-Pauly [57]). *The Hilbert cube does not finite-level Borel embed into $\mathbb{G}_\omega$, and $\mathbb{G}_\omega$ does not finite-level Borel embed into any countable-dimensional Polish space.*

Despite its importance, there are only a few results on *computable topological dimension theory*. The very first step was taken by Kenny [51]. Effectivizations of the existence of a Henderson compactum and the Cantor manifold theorem are discussed in Kihara [52]. Recently, McNicholl and Rute took an important next step. They introduced the notion of a *uniform degree*, which is a generalization of truth-table degrees in the setting of computable metric spaces. Then, for instance, they proved that a point in $\mathbb{R}^2$ is contained in a computable arc if and only if it has an $\mathbb{R}$-uniform degree. The notion of a uniform degree is connected to various notions in topological dimension theory. We have the impression that the further development of the generalized truth-table degrees would give new insights into the computability-theoretic nature of topological dimension theory.

2.8 Computability of Semicomputable Sets

Each computable closed set (and in particular computable) set is clearly co-c.e. closed. Conversely, a co-c.e. closed set need not be computable closed. Moreover, as noted in Section 2.4, there exists a nonempty co-c.e. subset of $\mathbb{R}$ which does not contain a computable point. Hence, co-c.e. sets can be "far away from being computable". However, it turns out that under certain assumptions we can conclude that a co-c.e. closed set is computable. The pioneering work in this area was done by Miller.

Theorem 2.8.1 (Miller [68]).

1. *If $S \subseteq \mathbb{R}^m$ is co-c.e. closed and $S \cong \mathbb{S}^n$ for some $n \geq 1$, then S is computable.*
2. *If $D \subseteq \mathbb{R}^m$ is co-c.e. closed and there exists, for some $n \geq 1$, a homeomorphism $f : \mathbb{B}^n \to D$ such that $f(\mathbb{S}^{n-1})$ is also co-c.e. closed, then D is computable.*

Here $\mathbb{S}^n$ denotes the unit sphere in $\mathbb{R}^{n+1}$, $\mathbb{B}^n$ denotes the unit closed ball in $\mathbb{R}^n$, and $X \cong Y$ denotes that topological spaces X and Y are homeomorphic.

By Theorem 2.8.1, in Euclidean space each co-c.e. closed topological circle is computable and each co-c.e. arc with computable endpoints is computable. The second claim of Theorem 2.8.1 does not hold in general if we omit the assumption that $f(\mathbb{S}^{n-1})$ is co-c.e. closed. At the same time, the fact that $f(\mathbb{S}^{n-1})$ is co-c.e. closed is not actually necessary for $f(\mathbb{B}^n)$ to be computable.

Theorem 2.8.2 (Miller [68]).

1. *There is a co-c.e. arc in $\mathbb{R}^2$ which is not computable.*
2. *There is a computable arc in $\mathbb{R}^2$ with non-computable endpoints.*

So, although a co-c.e. closed set need not be computable, it makes sense to ask the following general question: under what conditions does the implication

$$S \text{ co-c.e. closed} \implies S \text{ computable closed} \tag{2.7}$$

hold in a computable metric space (X,d,α)? Miller's work shows that topology plays an important role regarding conditions under which (2.7) holds. In Theorem 2.8.1, however, the ambient space is Euclidean space. It was later shown by Iljazović [39] that the claim of that theorem holds in more general computable metric spaces.

Theorem 2.8.3. *Let (X,d,α) be a computable metric space which has the effective covering property and compact closed balls. Suppose $S \subseteq X$ is such that $S \cong \mathbb{S}^n$ for some $n \geq 1$ or $S \cong \mathbb{B}^n$ by a homeomorphism $f : \mathbb{B}^n \to S$ such that $f(\mathbb{S}^{n-1})$ is a co-c.e. closed set in (X,d,α). Then implication (2.7) holds.*

In particular, (2.7) holds if S is a topological circle or an arc with computable endpoints (in a computable metric space which has the effective covering property and compact closed balls).

2.8.1 Semicomputable Chainable and Circularly Chainable Continua

Arcs and topological circles are just representatives of more general topological spaces: chainable and circularly chainable continua [74].

Let (X,d) be a metric space, $\varepsilon > 0$ and $C_0,\ldots,C_m$ a finite sequence of nonempty open sets in (X,d) such that $\operatorname{diam} C_i < \varepsilon$ for each $i \in \{0,\ldots,m\}$. We say that $C_0,\ldots,C_m$ is an *ε-chain* in (X,d) if, for all $i,j \in \{0,\ldots,m\}$, $C_i \cap C_j = \emptyset$ if and

only if $1 < |i-j|$. We say that $C_0,\dots,C_m$ is an *ε-circular chain* in (X,d) if, for all $i,j \in \{0,\dots,m\}$, $C_i \cap C_j = \emptyset$ if and only if $1 < |i-j| < m$. We say that a finite sequence of sets $C_0,\dots,C_m$ covers a set X if $X \subseteq C_0 \cup \dots \cup C_m$.

A continuum (X,d) will be called a *chainable continuum* if for each $\varepsilon > 0$ there exists an ε-chain in (X,d) which covers X. A continuum (X,d) will be called a *circularly chainable continuum* if for each $\varepsilon > 0$ there exists an ε-circular chain in (X,d) which covers X. If (X,d) is a continuum and $a,b \in X$, we say that (X,d) is a *continuum chainable from a to b* if for each $\varepsilon > 0$ there exists an ε-chain $C_0,\dots,C_m$ in (X,d) which covers X and such that $a \in C_0$ and $b \in C_m$.

The segment $[0,1]$ is a continuum chainable from 0 to 1. Consequently, if X is an arc with endpoints a and b, then X is a continuum chainable from a to b. The unit circle $\mathbb{S}^1$ is a circularly chainable continuum and therefore each topological circle is also a circularly chainable continuum. On the other hand, the space $K = (\{0\} \times [-1,1]) \cup \{(x, \sin\frac{1}{x}) \mid x \in (0,1]\}$, known as the topologist's sine curve, is an example of a chainable continuum (K is chainable from a to c and also from b to c, where $a = (0,-1)$, $b = (0,1)$ and $c = (1,\sin 1)$) which is not an arc. Furthermore, the space $W = K \cup (\{0\} \times [-2,-1]) \cup ([0,1] \times \{-2\}) \cup (\{1\} \times [-2,\sin 1])$, known as the Warsaw circle, is an example of a circularly chainable continuum which is not a topological circle.

Theorem 2.8.4 (Iljazović [37]). *Let (X,d,α) be a computable metric space which has the effective covering property and compact closed balls. Let $S \subseteq X$.*

1. *If S is (as a subspace of (X,d)) a circularly chainable continuum which is not chainable, then (2.7) holds.*
2. *If S is a continuum chainable from a to b, where a and b are computable points in (X,d,α), then (2.7) holds.*
3. *Suppose S is a co-c.e. closed set. If S is a chainable and* decomposable *continuum, then for each $\varepsilon > 0$ there exists a subcontinuum K of S such that K is computable and $d_H(S,K) < \varepsilon$. Moreover, K can be chosen so that it is chainable from a to b, where a and b are computable points.*

That a continuum K is *decomposable* means that there exist proper subcontinua K_1 and K_2 of K such that $K = K_1 \cup K_2$. For example, $[0,1]$ is decomposable since $[0,1] = [0,\frac{1}{2}] \cup [\frac{1}{2},1]$.

Each topological circle is a circularly chainable continuum which is not chainable. Therefore, Theorem 2.8.4 is a generalization of Theorem 2.8.3 for $n = 1$.

In Theorem 2.8.3 and Theorem 2.8.4 we assume that the computable metric space (X,d,α) has the effective covering property and compact closed balls. The natural question is: are these additional assumptions on a computable metric space necessary? The answer is affirmative: we cannot omit these assumptions. By [40] there exists a computable metric space such that the following hold:

1. There exists a co-c.e. closed topological circle S in (X,d,α) which does not contain a computable point (and which in particular is not computable closed);
2. There exists a co-c.e. closed arc S in (X,d,α) which is chainable from a to b, where a and b are computable points, but which is not computable closed;

3. There exists a co-c.e. closed arc S in (X,d,α) which does not contain a computable point (and which in particular cannot be approximated by a computable subcontinuum).

Moreover, (X,d,α) can be chosen so that either it has compact closed balls (but not the effective covering property) or it has the effective covering property (but not compact closed balls).

Recall that even a one-point co-c.e. closed set need not be computable (the discussion after Proposition 2.3.3). This indicates that we have to restrict ourselves to some special computable metric spaces if we are looking for topological conditions under which (2.7) holds.

On the other hand, by Proposition 2.3.6, in computable metric spaces which have the effective covering property and compact closed balls the conditions under which (2.7) holds are the same as the conditions under which the implication

$$S \text{ semicomputable} \implies S \text{ computable} \tag{2.8}$$

holds. It turns out, however, that it is more convenient to search for conditions under which (2.8) holds than for conditions under which (2.7) holds since we do not need any additional assumptions on the ambient space. For example, (2.8) holds in any computable metric space if $S \cong \mathbb{S}^n$ or $S \cong \mathbb{B}^n$ by a homeomorphism $f : \mathbb{B}^n \to S$ such that $f(\mathbb{S}^{n-1})$ is a semicomputable set [41]. This is a generalization of Theorem 2.8.3. Similarly, if in Theorem 2.8.4 we remove the assumptions on the computable metric space, replace "co-c.e. set" by "semicomputable set" and replace (2.7) by (2.8), we get a claim which also holds [43] and which generalizes Theorem 2.8.4.

Actually, we have a much more general result than Theorem 2.8.3: (2.8) holds if S is a compact manifold with computable boundary.

2.8.2 Semicomputable Manifolds

Let $\mathbb{H}^n = \{(x_1,\dots,x_n) \in \mathbb{R}^n \mid x_n \geq 0\}$ and $\operatorname{Bd}\mathbb{H}^n = \{(x_1,\dots,x_n) \in \mathbb{R}^n \mid x_n = 0\}$ (for $n \geq 1$).

A second-countable Hausdorff space X is said to be an *n–manifold with boundary* if for each $x \in X$ there exists a neighborhood N of x in X such that N is homeomorphic to $\mathbb{R}^n$ or there exists a homeomorphism $f\colon \mathbb{H}^n \to N$ such that $x \in f(\operatorname{Bd}\mathbb{H}^n)$.

If X is an n–manifold with boundary, we define ∂X to be the set of all points $x \in X$ which have no neighborhood homeomorphic to $\mathbb{R}^n$. We say that ∂X is the *boundary* of the manifold X.

If X is an n-manifold with boundary such that $\partial X = \emptyset$, then we say that X is an *n-manifold*. Hence a second-countable Hausdorff space X is an n-manifold if and only if each point $x \in X$ has a neighborhood homeomorphic to $\mathbb{R}^n$.

Let $n \in \mathbb{N}$. Then $\mathbb{S}^n$ is an n-manifold and $\mathbb{B}^n$ is an n-manifold with boundary; its boundary is $\mathbb{S}^{n-1}$ [72]. Consequently, if $X \cong \mathbb{S}^n$, then X is an n-manifold. Further-

more, if $f:\mathbb{B}^n \to X$ is a homeomorphism, then X is an n-manifold with boundary and $\partial X = f(\mathbb{S}^{n-1})$.

If (X,d) is a metric space, $A \subseteq X$, and $r>0$, then we will denote by $N_r(A)$ the *r-neighborhood* of A, i.e.

$$N_r(A) = \bigcup_{x\in A} B(x,r).$$

Note that for $A,B \subseteq X$ and $r>0$ we have that A and B are r-close if and only if $A \subseteq N_r(B)$ and $B \subseteq N_r(A)$. The following notion is useful in the proof of the fact that (2.8) holds for compact manifolds with computable boundaries.

Let (X,d,α) be a computable metric space and $A,B \subseteq X$, $A \subseteq B$. We say that A is *computable up to* B if there exists a computable function $f:\mathbb{N}\to\mathbb{N}$ such that

$$A \subseteq N_{2^{-k}}(\Lambda_{f(k)}) \text{ and } \Lambda_{f(k)} \subseteq N_{2^{-k}}(B)$$

for each $k \in \mathbb{N}$ (recall definition (2.1)). Clearly, if S is a nonempty compact set in (X,d), then S is computable if and only if S is computable up to S. Furthermore, it is easy to prove the following fact (see [41]): if $A_1,\dots,A_n$ are sets computable up to S, then $A_1 \cup \cdots \cup A_n$ is computable up to S.

If $S \subseteq X$ and $x \in S$, we say that S is *computable at* x if there exists a neighborhood N of x in S such that N is computable up to S. The proof of the following proposition is straightforward (see [41]).

Proposition 2.8.5. *Let (X,d,α) be a computable metric space and let $S \subseteq X$ be a compact set. Then S is computable if and only if S is computable at x for each $x \in S$.*

A connection between topology and computability is apparent in the following result: if x has a Euclidean neighborhood in S, then S is computable at x.

Theorem 2.8.6 (Iljazović [41]). *Let (X,d,α) be a computable metric space and let S be a semicomputable set in this space.*

1. *Suppose $x \in S$ is a point which has a neighborhood in S homeomorphic to $\mathbb{R}^n$ for some $n \in \mathbb{N}\setminus\{0\}$. Then S is computable at x.*
2. *Let T be a semicomputable set such that $T \subseteq S$. Suppose $x \in S$ is a point which has a neighborhood N in S with the following property: there exists $n \in \mathbb{N}\setminus\{0\}$ and a homeomorphism $f:\mathbb{H}^n \to N$ such that $f(\mathrm{Bd}\,\mathbb{H}^n) = N \cap T$. Then S is computable at x.*

Actually, the original result from [41] assumes that S is compact, but this assumption can easily be removed (see Theorem 5.2 in [46]).

Proposition 2.8.5 and Theorem 2.8.6 imply that (2.8) holds if S is a compact manifold with computable boundary. This can be stated in the following way.

Theorem 2.8.7 ([41]). *Let (X,d,α) be a computable metric space and let S be a semicomputable set in this space which is, as a subspace of (X,d), a compact manifold with boundary. Then the following implication holds:*

$$\partial S \text{ computable } \Longrightarrow S \text{ computable}.$$

In particular, each semicomputable compact manifold is computable.

In general, $S \subseteq \mathbb{R}^n$ is co-c.e. closed if and only if $S = f^{-1}(\{0\})$ for some computable function $f : \mathbb{R}^n \to \mathbb{R}$ [12]. Hence, if $f : \mathbb{R}^n \to \mathbb{R}$ is a computable function, then $f^{-1}(\{0\})$ need not be a computable set, moreover $f^{-1}(\{0\})$ need not contain a computable point even if $f^{-1}(\{0\}) \neq \emptyset$.

Let $f : \mathbb{R}^n \to \mathbb{R}^m$ be a function of class C^1. Suppose $y \in \mathbb{R}^m$ is a regular value of f (which means that the differential $D(f)(x) : \mathbb{R}^n \to \mathbb{R}^m$ of f in x is surjective for each $x \in f^{-1}(\{y\})$) and $f^{-1}(\{y\}) \neq \emptyset$. Then it is known from differential topology (see, e.g. [85]) that $f^{-1}(\{y\})$ is an $(n-m)$-manifold. Additionally, if f is computable, then $f^{-1}(\{y\})$ is co-c.e. closed and therefore semicomputable (Proposition 2.3.6). The following is a consequence of Theorem 2.8.7.

Corollary 2.8.8. *If $f : \mathbb{R}^n \to \mathbb{R}^m$ is a computable function of class C^1 and $y \in \mathbb{R}^m$ is a regular value of f such that $f^{-1}(\{y\})$ is bounded, then $f^{-1}(\{y\})$ is computable.*

For example, Corollary 2.8.8 easily implies that the set of all $(x,y,z) \in \mathbb{R}^3$ such that $x^2(1+e^x)+y^2(1+e^y)+z^2(1+e^z) = 1$ is computable (see [41]).

The claim of Theorem 2.8.7 need not hold if we omit the assumption that S is compact. This is shown by the following simple example from [15].

Example 2.8.9. Let A be a c.e. subset of $\mathbb{N}$ which is not computable. Then the set $B = \mathbb{N} \setminus A$ is co-c.e. closed in $\mathbb{R}$ and therefore the set $S = B \times \mathbb{R}$ is co-c.e. closed in $\mathbb{R}^2$. Since $B \subseteq \mathbb{N}$, S is a 1-manifold. So S is a semicomputable 1-manifold in $\mathbb{R}^2$, but S is not computable in $\mathbb{R}^2$, which can be deduced from the fact that B is not computable in $\mathbb{N}$.

In general, if X is an n-manifold with boundary, then each connected component of X is also an n-manifold with boundary. Semicomputable 1-manifolds with boundaries have been studied in [15]. It is a well known fact (see e.g. [85]) that if X is a connected 1-manifold with boundary, then X is a topological line (i.e. $X \cong \mathbb{R}$) or a topological ray (i.e. $X \cong [0,\infty)$) or a topological circle or an arc.

Theorem 2.8.10 (Burnik-Iljazović [15]). *Let (X,d,α) be a computable metric space and let S be a semicomputable set in this space which is, as a subspace of (X,d), a 1-manifold with boundary. Let K be a connected component of S.*

1. *If K is a topological line or circle, then K is c.e. closed in (X,d,α).*
2. *If K is a topological ray with computable endpoint or an arc with computable endpoints, then K is c.e. closed.*
3. *If ∂S is semicomputable, then each connected component of S is c.e. closed.*

Since the union of finitely many c.e. closed sets is c.e. closed, an immediate consequence of Theorem 2.8.10 is the following theorem.

Theorem 2.8.11 ([15]). *Let (X,d,α) be a computable metric space and let S be a semicomputable set in this space which is a 1-manifold with boundary. Suppose that S has finitely many connected components. Then the following implication holds:*

$$\partial S \text{ computable} \implies S \text{ computable}.$$

In particular, each semicomputable 1*-manifold with finitely many connected components is computable.*

It should be mentioned that the uniform versions of Theorems 2.8.3, 2.8.7 and 2.8.11 do not hold in general: there exists a sequence of topological circles in $\mathbb{R}^2$ (in fact in $[0,1]^2$) which is uniformly semi-computable, but not uniformly computable (Example 7 in [37]).

If S is a co-c.e. closed set in a computable metric space (X,d,α) such that $X \setminus S$ is disconnected (for example, this holds by the Generalized Jordan Curve Theorem if S is homeomorphic to $\mathbb{S}^n$ and $X = \mathbb{R}^{n+1}$), then it is possible to conclude that, under some additional assumptions, S is computable closed or at least contains a computable point. Such conditions have been studied in [37, 42].

Finally, let us mention that semicomputable manifolds in *computable topological spaces* have been studied in [44]. For a study of computable topological spaces see [98, 96].

2.8.3 Inner Approximation

Let $\mathscr{A}$ be a collection of continua, and let B be a continuum. We say that B *is inner approximated by* $\mathscr{A}$ if, for any $\varepsilon > 0$, there exists $A \in \mathscr{A}$ such that $A \subseteq B$ and $d_H(A,B) < \varepsilon$, where we recall that d_H is the Hausdorff distance. Classically, every arcwise-connected continuum is inner approximated by locally connected continua.

In the computability-theoretic setting, claim 3 of Theorem 2.8.4 says the following: every co-c.e. chainable and decomposable continuum is inner approximated by computable chainable continua (in appropriate computable metric spaces; in fact, by [43], in any computable metric space any semicomputable chainable and decomposable continuum is inner approximated by computable chainable continua).

However, we do not always have a computable inner approximation. Recall that X is contractible if the identity map on X is null-homotopic, and that a curve means a one-dimensional continuum.

Theorem 2.8.12 (Kihara [53]).

1. *There exists a contractible, locally contractible, co-c.e., planar curve which is not inner approximated by computable continua.*
2. *There exists a contractible, computable, planar curve which is not inner approximated by locally connected, co-c.e. continua.*

2.8.4 Density of Computable Points in Semicomputable Sets

Let (X,d,α) be a computable metric space, $S \subseteq X$, and $x \in S$. Suppose that there exists a neighborhood N of x in S such that N is computable in (X,d,α). Then N, as

a subset of S, is clearly computable up to S. So if x has a computable neighborhood in S, then S is computable at x.

Theorem 2.8.13 (Iljazović-Validžić [46]). *Let (X,d,α) be a computable metric space and S a complete set in (X,d). Suppose S is computable at x. Then there exists a neighborhood N of x in S such that N is computable compact in (X,d,α). Moreover, for each $\varepsilon > 0$ there exists such an N with the property that* $\operatorname{diam} N < \varepsilon$.

If S is a nonempty compact set in a metric space (X,d) and $\varepsilon > 0$, then $S \cap \overline{B}(x, \frac{\varepsilon}{3})$ is a compact set for each $x \in X$, and it follows readily that there exist compact sets $K_1, \ldots, K_n$ whose union is S and whose diameters are less than ε. On the other hand, if S is a computable compact set, $i \in \mathbb{N}$, and $r \in \mathbb{Q}$, $r > 0$, then the intersection $S \cap \overline{B}(\alpha_i, r)$ need not be computable compact even if $\overline{B}(\alpha_i, r)$ is computable compact (see [46]). Nevertheless, we have the following result.

Corollary 2.8.14 ([46]). *Let (X,d,α) be a computable metric space and let S be a nonempty computable compact set in this space. Then for each $\varepsilon > 0$ there exist nonempty computable compact sets $K_1, \ldots, K_n$ such that $S = K_1 \cup \cdots \cup K_n$ and* $\operatorname{diam} K_i < \varepsilon$ *for each $i \in \{1, \ldots, n\}$.*

Proof. Let $\varepsilon > 0$. Since S is computable, it is computable at each point $x \in S$ and therefore, by Theorem 2.8.13, each point of S has a neighborhood in S which is computable and has the diameter less than ε. This, together with the compactness of S, proves the claim of the corollary. □

Using Corollary 2.8.14 it is easy to deduce the following facts (see [46]): if (F,G) is a separation of a computable compact set, then F and G are computable compact sets; if S is a computable compact set which has finitely many components, then each of these components is a computable compact set.

A connected component of a computable compact set need not be computable in general. For example, there are uncountably many components of the Cantor set, so at least one of them is not computable.

Let S be a semicomputable set in a computable metric space. Then S may not be computable but, at the same time, computable points which lie in S may be dense in S. Therefore, the general question is under what conditions the implication

$$S \text{ semicomputable} \implies \text{computable points are dense in } S \tag{2.9}$$

holds in a computable metric space. If (2.8) holds under certain conditions, then under the same conditions (2.9) also holds. The converse does not hold. For example, (2.9) holds if S is an arc in Euclidean space, but (2.8) need not hold if S is an arc in Euclidean space [68]. Miller has proved in [68] that (2.9) holds in Euclidean space if $S \cong \mathbb{B}^n$ for some $n \in \mathbb{N}$.

The following result follows from Theorem 2.8.6 and Theorem 2.8.13.

Theorem 2.8.15 (Iljazović-Validžić [46]). *Let (X,d,α) be a computable metric space and let S be a semicomputable set in this space. Suppose that S is a manifold with boundary or S has the topological type of a polyhedron. Then the set of all computable points which belong to S is dense in S.*

That S has the topological type of a polyhedron means that $S \cong P$ for some polyhedron P. A polyhedron is a space obtained from simplices (line segments, triangles, tetrahedra, and their higher-dimensional analogues) by gluing them together along their faces (see [72] for the definition).

Suppose S is a nonempty compact set in which computable points are dense. Then for each $\varepsilon > 0$ there exist computable points $x_0, \dots, x_n \in S$ such that S and $\{x_0, \dots, x_n\}$ are ε-close. So, for each $\varepsilon > 0$ there exists a computable set K such that $K \subseteq S$ and $d_H(S,K) < \varepsilon$. In particular, if S is a nonempty semicomputable compact set which satisfies the conditions of Theorem 2.8.15, then S can be approximated by its computable subset with arbitrary precision. However, if S is a nonempty semicomputable compact manifold with boundary, then we can find an even better approximation, namely for each $\varepsilon > 0$ there exists a computable subset K of S which is ε-close to S and covers the entire set S except for some part of S which lies in an ε-neighborhood of ∂S (see Theorem 5.4 in [46]).

At the end of this section, let us mention the following problem. Suppose (X,d,α) is a computable metric space and U and V are c.e. open sets in this space. Let $S = X \setminus (U \cup V)$. Suppose A is a computable compact set which is connected and which intersects both U and V. Then clearly $A \cap S \neq \emptyset$. The question is: does $A \cap S$ have to contain a computable point? It is not hard to see that the answer in general is negative [43]. An affirmative answer to this question is given in [43] in the case when A is an arc and, under some additional assumptions, in the case when A is a chainable continuum. These results can be considered as generalizations of the computable intermediate value theorem: if $f : [0,1] \to \mathbb{R}$ is a computable function such that $f(0) < 0$ and $f(1) > 0$, then f has a computable zero-point [82].

Let K be the unit square in the plane, i.e., $K = [0,1] \times [0,1]$, and let $\mathring{K}$ be the corresponding open unit square. Suppose $f, g : [0,1] \to K$ are continuous functions such that $f(0) = (0,0)$, $f(1) = (1,1)$, $g(0) = (0,1)$, and $g(1) = (1,0)$. Then the images of f and g intersect. This nontrivial fact can be proved using the Jordan curve theorem (see e.g. [97]). Manukyan has proved that a constructive version of this result does not hold in general (even with the assumption that $f(t), g(t) \in \mathring{K}$ for each $t \in [0,1] \setminus \{0,1\}$) [63, 59]. The open question was: does a computable version of this result hold? Weihrauch has recently solved this problem; his result can also be considered to be a generalization of the computable intermediate value theorem.

Theorem 2.8.16 (Weihrauch [97]). *Let $f, g : [0,1] \to K$ be computable functions such that $f(0) = (0,0)$, $f(1) = (1,1)$, $g(0) = (0,1)$, and $g(1) = (1,0)$. Then the images of f and g intersect in a computable point.*

The image of a computable function $[0,1] \to \mathbb{R}^n$ is easily seen to be a computable compact set. On the other hand, the intersection of a semicomputable set and a co-c.e. closed set is a semicomputable set (in any computable metric space, see [43]). So Theorem 2.8.16 and the above results from [43] can be viewed as results which provide conditions under which a semicomputable set contains a computable point.

2.9 Computable Images of a Segment

As said, if $f : [0,1] \to \mathbb{R}^n$ is a computable function, then $f([0,1])$ is a computable compact set in $\mathbb{R}^n$. The question is what can be said about various forms of the converse of this statement. By Example 2.6.1, there exists a computable arc A in $\mathbb{R}^2$ with computable endpoints such that A is not the image of any computable injection $[0,1] \to \mathbb{R}^2$ (in fact, A is the image of a computable function $[0,1] \to \mathbb{R}^2$). Moreover, we have the following result.

Theorem 2.9.1 (Gu-Lutz-Mayordomo [30]). *There exists an arc A in $\mathbb{R}^2$ with the following properties:*

1. *A is rectifiable (i.e. A has finite length) and smooth except at one endpoint;*
2. *there exists a computable function $f : [0,1] \to \mathbb{R}^2$ whose image is A (moreover, f, the velocity function f', and the acceleration function f'' are polynomial-time computable);*
3. *for any computable function $f : [0,1] \to \mathbb{R}^2$ whose image is A and for every $m \in \mathbb{N}$, $m \geq 1$, there exist disjoint closed subintervals $I_0, \ldots, I_m$ of $[0,1]$ such that the arc $f(I_0)$ has positive length and $f(I_i) = f(I_0)$ for each $i \in \{1, \ldots, m\}$.*

If $f : [0,1] \to \mathbb{R}^2$ is a computable injection, then the arc $f([0,1])$ need not be rectifiable. On the other hand, if $f : [0,1] \to \mathbb{R}^2$ is a computable function such that the curve $f([0,1])$ is rectifiable, then $f([0,1])$ clearly need not be an arc. The following result points out a significant difference between these two types of sets.

Theorem 2.9.2 (McNicholl [64]).

1. *There exists a computable injection $f : [0,1] \to \mathbb{R}^2$ and a point $x \in f([0,1])$ such that there exists no computable function $g : [0,1] \to \mathbb{R}^2$ with the following property: $g([0,1])$ is a rectifiable curve and $x \in g([0,1])$.*
2. *There exists a computable function $g : [0,1] \to \mathbb{R}^2$ such that $g([0,1])$ is a rectifiable curve and a point $x \in g([0,1])$ with the following property: there exists no computable injection $f : [0,1] \to \mathbb{R}^2$ such that $x \in f([0,1])$.*

Of course, the arc $f([0,1])$ in 1. cannot be rectifiable and the function g in 2. cannot be injective.

A metrizable space which is locally connected, connected, and compact is called a *Peano continuum*. The Hahn-Mazurkiewicz theorem (see e.g. [74, 18]) says that a Hausdorff space X is a Peano continuum if and only if there is a continuous surjection $[0,1] \to X$.

If X is a subset of the plane, it makes sense to ask whether a computable version of the Hahn-Mazurkiewicz theorem holds.

Theorem 2.9.3 (Couch-Daniel-McNicholl [20]). *There is a computable compact set X in $\mathbb{R}^2$ which is a Peano continuum and such that there exists no computable function $[0,1] \to \mathbb{R}^2$ whose image is X.*

However, the situation changes under the assumption that X is effectively locally connected. If $X \subseteq \mathbb{R}^n$, a *local connectivity operator* for X is a continuous operator

that, given a name of a point $p \in X$ in $\mathbb{R}^n$ and a name of a rational rectangle R in $\mathbb{R}^n$ which contains p gives a name of an open set U in $\mathbb{R}^n$ such that $U \cap X$ is connected and $p \in U \cap X \subseteq R$ (see [20]). The set X is defined in [20] to be *effectively locally connected* if it has a computable local connectivity operator.

Theorem 2.9.4 (Couch-Daniel-McNicholl [20]). *Let $X \subseteq \mathbb{R}^n$ be a computable Peano continuum. Suppose X is effectively locally connected. Then there exists a computable function $g : [0,1] \to \mathbb{R}^n$ whose image is X. Moreover, a name of such a function g can be uniformly computed from a name of a computable compact set X and a name of a local connectivity operator for X.*

The following theorem is a computable version of another well-know topological result: each compact metric space is a continuous image of the Cantor set (see Theorem 6.C.12 in [18]).

Theorem 2.9.5 (Couch-Daniel-McNicholl [20]). *Let C be the Cantor middle-third set. Let X be a nonempty computable compact set in $\mathbb{R}^n$. Then there exists a computable surjection $C \to X$. Moreover, a name of such a surjection can be uniformly computed from a name of a computable compact set X.*

2.10 Computability Structures

If (X,d) is a metric space and α a sequence in X such that (X,d,α) is a computable metric space, then we say that α is an *effective separating sequence* in (X,d) [102]. Suppose α and β are effective separating sequences in a metric space (X,d). We say that α and β are *equivalent* if α is a computable sequence in (X,d,β) and β is a computable sequence in (X,d,α).

If (X,d,α) is a computable metric space, let $\mathscr{S}_\alpha$ denote the set of all computable sequences in (X,d,α). It is easy to conclude that effective separating sequences α and β in (X,d) are equivalent if and only if $\mathscr{S}_\alpha = \mathscr{S}_\beta$ [45].

The notions of a computable point, a computable sequence, a c.e. closed set, a co-c.e. closed set, and a computable compact set in a computable metric space (X,d,α) depend, by definition, on the sequence α. However, it is easy to conclude that these notions coincide in (X,d,α) and (X,d,β) if α and β are equivalent effective separating sequences. This means that these notions can be viewed as notions defined related to the entire set $\mathscr{S}_\alpha$ and not just to α itself. Therefore, we can take sets of the form $\mathscr{S}_\alpha$ as a basis for computability concepts on a metric space (X,d), which leads to the notion of a computability structure on a metric space.

A *computability structure* $\mathscr{S}$ on a metric space (X,d) is a set of sequences in X such that the following hold [70, 102, 82, 45]:

1. if $(x_i),(y_j) \in \mathscr{S}$, then the function $\mathbb{N}^2 \to \mathbb{R}$, $(i,j) \mapsto d(x_i,y_j)$, is computable;
2. if $(x_i) \in \mathscr{S}$ and (y_i) is a sequence in X such that $d(y_i,x_{F(i,k)}) < 2^{-k}$ for all $i,k \in \mathbb{N}$, where $F : \mathbb{N}^2 \to \mathbb{N}$ is a computable function, then $(y_i) \in \mathscr{S}$.

A computability structure $\mathscr{S}$ on a metric space (X,d) is said to be *separable* if there exists $(x_i) \in \mathscr{S}$ such that (x_i) is a dense sequence in (X,d).

If (X,d) is a metric space, then there exist computability structures on (X,d): we can take any $a \in X$, and then $\{(a,a,a,\dots)\}$ is trivially a computability structure on (X,d). On the other hand, a separable computability structure on a metric space (X,d) need not exist. It certainly does not exist if (X,d) is not a separable metric space, but even if (X,d) is separable, a separable computability structure on (X,d) need not exist. For example, we can take $X = \{0,\gamma\}$, where γ is an incomputable real number, and the Euclidean metric d on X.

The general question is: if (X,d) is a metric space, how many separable computability structures exist on (X,d)?

A computable metric space (X,d,α) is said to be *effectively totally bounded* if there exists a computable function $f : \mathbb{N} \to \mathbb{N}$ such that

$$X = B(\alpha_0, 2^{-k}) \cup \cdots \cup B(\alpha_{f(k)}, 2^{-k})$$

for each $k \in \mathbb{N}$. If (X,d,α) is effectively totally bounded, then (X,d) is obviously totally bounded. Conversely, if (X,d) is totally bounded, (X,d,α) need not be effectively totally bounded [38].

If α and β are equivalent effective separating sequences on a metric space (X,d,α), then it is not hard to conclude that (X,d,α) is effectively totally bounded if and only if (X,d,β) is effectively totally bounded. However, this claim holds even if α and β are not equivalent.

Theorem 2.10.1 (Iljazović [38]). *If α and β are effective separating sequences in a metric space (X,d), then (X,d,α) is effectively totally bounded if and only if (X,d,β) is effectively totally bounded.*

Using Theorem 2.3.4 and Proposition 2.3.6 it is easy to conclude that a computable metric space (X,d,α) is effectively compact if and only if (X,d) is compact and (X,d,α) is effectively totally bounded.

Theorem 2.10.2 (Iljazović [38]). *Let (X,d,α) be an effectively compact computable metric space such that there exist only finitely many isometries of the metric space (X,d). Then there exists a unique separable computability structure on (X,d).*

For example, Theorem 2.10.2 implies that there exists a unique separable computability structure on $[0,1]$.

Let (X,d) be a metric space, $\mathscr{S}$ a set of sequences in X, and $f : X \to X$ an isometry. Let $f(\mathscr{S})$ denote the set $\{(f(x_i)) \mid (x_i) \in \mathscr{S}\}$. Then $\mathscr{S}$ is a (separable) computability structure on (X,d) if and only if $f(\mathscr{S})$ is a (separable) computability structure on (X,d).

A metric space (X,d) is said to be *computably categorical* if for all separable computability structures $\mathscr{S}$ and $\mathscr{T}$ on (X,d) there exists an isometry $f : X \to X$ such that $f(\mathscr{S}) = \mathscr{T}$ [66].

Theorem 2.10.3 (Melnikov [66]).

1. Every separable Hilbert space is computably categorical (as a metric space).

2. *The space $C[0,1]$ of all continuous functions $[0,1] \to \mathbb{R}$ with the metric of uniform convergence (supremum metric) is not computably categorical.*
3. *The Cantor space $\{0,1\}^{\mathbb{N}}$ with the metric $d((x_n),(y_n)) = \max\{2^{-n} \mid x_n \neq y_n\}$ is computably categorical.*

Theorem 2.10.4 (McNicholl [65]). *Let $p \in \mathbb{R}$, $p \geq 1$. Let l^p be the set of all sequences (x_i) of complex (or real) numbers such that $\sum_{i=0}^{\infty} |x_i|^p < \infty$. Then l^p, with the metric induced by the norm $\|(x_i)\| = (\sum_{i=0}^{\infty} |x_i|^p)^{\frac{1}{p}}$, is computably categorical if and only if $p = 2$.*

The general question is: if (X,d) is a separable metric space, does there exist a metric space (Y,d') which is homeomorphic to (X,d) and which has a separable computability structure? A similar question is this: if S is a (compact) set in a computable metric space, does there exist a computable compact set T in the same space such that S and T are homeomorphic? Bosserhoff and Hertling have studied a similar problem in Euclidean space and they got the following result.

Theorem 2.10.5 (Bosserhoff-Hertling [3]). *Let $n \in \mathbb{N}$, $n \geq 1$.*

1. *There exists a c.e. closed set K in $\mathbb{R}^n$ such that K is compact, $K \subseteq [0,1]^n$, and such that $f(K)$ is not a computable compact set in $\mathbb{R}^n$ for any homeomorphism $f : \mathbb{R}^n \to \mathbb{R}^n$.*
2. *There exists a co-c.e. closed set K in $\mathbb{R}^n$ such that K is compact, $K \subseteq [0,1]^n$, and such that $f(K)$ is not a computable compact set in $\mathbb{R}^n$ for any homeomorphism $f : \mathbb{R}^n \to \mathbb{R}^n$.*

A computability structure on a metric space is called *maximal* if it is maximal with respect to inclusion. Each separable computability structure is maximal, but the converse does not hold in general. The question is under what conditions a maximal computability structure is separable. Another question is under what conditions a maximal computability structure on a metric space is unique.

Let $\mathscr{S}$ be a computability structure on a metric space (X,d) and let $a \in X$. We say that a is a *computable point* in $\mathscr{S}$ if there exists $(x_i) \in \mathscr{S}$ and $i \in \mathbb{N}$ such that $x_i = a$.

For $X \subseteq \mathbb{R}^n$, $X \neq \emptyset$, let $\dim X$ be the largest number $k \in \mathbb{N}$ such that there exist geometrically independent points $a_0, \ldots, a_k \in X$.

Theorem 2.10.6 (Iljazović-Validžić [46]).

1. *Each maximal computability structure on $\mathbb{R}^n$ is separable.*
2. *If $X \subseteq \mathbb{R}^n$, $\dim X = k$, $k \geq 1$, and $a_0, \ldots, a_{k-1} \in X$ are geometrically independent points such that $d(x_i,x_j)$ is a computable number for all $i,j \in \{0,\ldots,k-1\}$, where d is the Euclidean metric on X, then there exists a unique maximal computability structure on (X,d) in which $a_0, \ldots, a_{k-1}$ are computable points.*
3. *Let $\gamma > 0$. For $a \in [0,\gamma]$ let $\mathscr{M}_a$ be the unique maximal computability structure on $[0,\gamma]$ in which a is a computable point (such a computability structure exists by claim 2). Then $\mathscr{M}_a$ is a separable computability structure if and only if a and $\gamma - a$ are left computable numbers.*

References

1. Bauer, A.: The realizability approach to computable analysis and topology. School of Computer Science, Carnegie Mellon University, Pittsburgh (2000)
2. Bauer, A.: An injection from the Baire space to natural numbers. Mathematical Structures in Computer Science **25**, 1484–1489 (2015)
3. Bosserhoff, V., Hertling, P.: Effective subsets under homeomorphisms of R^n. Information and Computation **245**, 197–212 (2015)
4. Brattka, V.: Random numbers and an incomplete immune recursive set. In: P. Widmayer, F. Triguero, R. Morales, M. Hennessy, S. Eidenbenz, R. Conejo (eds.) Automata, Languages and Programming, *Lecture Notes in Computer Science*, vol. 2380, pp. 950–961. Springer, Berlin (2002), 29th International Colloquium, ICALP, Málaga, Spain, July 8–13, 2002
5. Brattka, V.: Computability over topological structures. In: S.B. Cooper, S.S. Goncharov (eds.) Computability and Models, pp. 93–136. Kluwer Academic Publishers, New York (2003)
6. Brattka, V.: Effective Borel measurability and reducibility of functions. Mathematical Logic Quarterly **51**(1), 19–44 (2005)
7. Brattka, V.: Plottable real number functions and the computable graph theorem. SIAM Journal on Computing **38**(1), 303–328 (2008)
8. Brattka, V., de Brecht, M., Pauly, A.: Closed choice and a uniform low basis theorem. Annals of Pure and Applied Logic **163**, 986–1008 (2012)
9. Brattka, V., Gherardi, G.: Borel complexity of topological operations on computable metric spaces. Journal of Logic and Computation **19**(1), 45–76 (2009)
10. Brattka, V., Hertling, P., Weihrauch, K.: A tutorial on computable analysis. In: S.B. Cooper, B. Löwe, A. Sorbi (eds.) New Computational Paradigms: Changing Conceptions of What Is Computable, pp. 425–491. Springer, New York (2008)
11. Brattka, V., Le Roux, S., Miller, J.S., Pauly, A.: Connected choice and the Brouwer fixed point theorem. Journal of Mathematical Logic **19**(1), 1–46 (2019)
12. Brattka, V., Presser, G.: Computability on subsets of metric spaces. Theoretical Computer Science **305**, 43–76 (2003)
13. Brattka, V., Weihrauch, K.: Computability on subsets of Euclidean space I: Closed and compact subsets. Theoretical Computer Science **219**, 65–93 (1999)
14. de Brecht, M.: Levels of discontinuity, limit-computability, and jump operators. In: V. Brattka, H. Diener, D. Spreen (eds.) Logic, Computation, Hierarchies, *Ontos Mathematical Logic*, pp. 93–122. Walter de Gruyter, Boston (2014)
15. Burnik, K., Iljazović, Z.: Computability of 1-manifolds. Logical Methods in Computer Science **10**(2), 2:8, 28 (2014)
16. Cenzer, D.: Π^0_1–classes in computability theory. In: E.R. Griffor (ed.) Handbook of Computability Theory, *Studies in Logic and the Foundations of Mathematics*, vol. 140, pp. 37–85. Elsevier, Amsterdam (1999)
17. Cenzer, D., Remmel, J.: Π^0_1–classes in mathematics. In: Y.L. Ershov, S. Goncharov, A. Nerode, J. Remmel, V. Marek (eds.) Handbook of Recursive Mathematics, *Studies in Logic and the Foundations of Mathematics*, vol. 139, pp. 623–821. Elsevier, Amsterdam (1998), volume 2, Recursive Algebra, Analysis and Combinatorics
18. Christenson, C.O., Voxman, W.L.: Aspects of Topology, *Pure and Applied Mathematics*, vol. 39. Marcel Dekker, New York-Basel (1977)
19. Clote, P.: On recursive trees with a unique infinite branch. Proceedings of the American Mathematical Society **93**(2), 335–342 (1985)
20. Couch, P.J., Daniel, B.D., McNicholl, T.H.: Computing space-filling curves. Theory of Computing Systems **50**(2), 370–386 (2012)
21. Day, A.R., Miller, J.S.: Randomness for non-computable measures. Transactions of the American Mathematical Society **365**(7), 3575–3591 (2013)
22. Diamondstone, D.E., Dzhafarov, D.D., Soare, R.I.: Π^0_1 classes, Peano arithmetic, randomness, and computable domination. Notre Dame Journal of Formal Logic **51**(1), 127–159 (2010)

23. Eršov, J.L.: Theory of numberings. In: E.R. Griffor (ed.) Handbook of Computability Theory, *Studies in Logic and the Foundations of Mathematics*, vol. 140, pp. 473–503. Elsevier, Amsterdam (1999)
24. Escardó, M.: Synthetic topology of data types and classical spaces. Electronic Notes in Theoretical Computer Science **87**, 21–165 (2004)
25. Galatolo, S., Hoyrup, M., Rojas, C.: Dynamics and abstract computability: computing invariant measures. Discrete and Continuous Dynamical Systems. Series A **29**(1), 193–212 (2011)
26. Gherardi, G.: Effective Borel degrees of some topological functions. Mathematical Logic Quarterly **52**(6), 625–642 (2006)
27. Gregoriades, V.: Classes of Polish spaces under effective Borel isomorphism. Memoirs of the American Mathematical Society **240**(1135), vii+87 (2016)
28. Gregoriades, V., Kihara, T., Ng, K.M.: Turing degrees in Polish spaces and decomposibility of Borel functions. Journal of Mathematical Logic **21**(01), 2050021 (2021)
29. Gregoriades, V., Kispéter, T., Pauly, A.: A comparison of concepts from computable analysis and effective descriptive set theory. Mathematical Structures in Computer Science **27**(8), 1414–1436 (2017)
30. Gu, X., Lutz, J.H., Mayordomo, E.: Curves that must be retraced. Information and Computation **209**(6), 992–1006 (2011)
31. Guthrie, J.A.: A characterization of $\aleph_0$-spaces. General Topology and its Applications **1**(2), 105–110 (1971)
32. Hertling, P.: A comparison of certain representations of regularly closed sets. In: V. Brattka, M. Schröder, K. Weihrauch (eds.) CCA 2002 Computability and Complexity in Analysis, *Electronic Notes in Theoretical Computer Science*, vol. 66. Elsevier, Amsterdam (2002), 5th International Workshop, CCA 2002, Málaga, Spain, July 12–13, 2002
33. Higuchi, K., Kihara, T.: Inside the Muchnik degrees I: Discontinuity, learnability and constructivism. Annals of Pure and Applied Logic **165**(5), 1058–1114 (2014)
34. Higuchi, K., Kihara, T.: Inside the Muchnik degrees II: The degree structures induced by the arithmetical hierarchy of countably continuous functions. Annals of Pure and Applied Logic **165**(6), 1201–1241 (2014)
35. Hinman, P.G.: A survey of Mučnik and Medvedev degrees. The Bulletin of Symbolic Logic **18**(2), 161–229 (2012)
36. Hurewicz, W., Wallman, H.: Dimension Theory, *Princeton Mathematical Series*, vol. 4. Princeton University Press, Princeton (1941)
37. Iljazović, Z.: Chainable and circularly chainable co-r.e. sets in computable metric spaces. Journal of Universal Computer Science **15**(6), 1206–1235 (2009)
38. Iljazović, Z.: Isometries and computability structures. Journal of Universal Computer Science **16**(18), 2569–2596 (2010)
39. Iljazović, Z.: Co-c.e. spheres and cells in computable metric spaces. Logical Methods in Computer Science **7**(3), 3:05, 21 (2011)
40. Iljazović, Z.: Local computability of computable metric spaces and computability of co-c.e. continua. Glas. Mat. Ser. III **47(67)**(1), 1–20 (2012)
41. Iljazović, Z.: Compact manifolds with computable boundaries. Logical Methods in Computer Science **9**(4), 4:19, 22 (2013)
42. Iljazović, Z., Pažek, B.: Co-c.e. sets with disconnected complements. Theory of Computing Systems **62**(5), 1109–1124 (2018)
43. Iljazović, Z., Pažek, B.: Computable intersection points. Computability **7**(3), 57–99 (2018)
44. Iljazović, Z., Sušić, I.: Semicomputable manifolds in computable topological spaces. Journal of Complexity **45**, 83–114 (2018)
45. Iljazović, Z., Validžić, L.: Maximal computability structures. Bulletin of Symbolic Logic **22**(04) (2016)
46. Iljazović, Z., Validžić, L.: Computable neighbourhoods of points in semicomputable manifolds. Annals of Pure and Applied Logic **168**(4), 840–859 (2017)
47. Jayne, J.E.: The space of class α Baire functions. Bulletin of the American Mathematical Society **80**, 1151–1156 (1974)

48. Jayne, J.E., Rogers, C.A.: First level Borel functions and isomorphisms. Journal de Mathématiques Pures et Appliquées. Neuvième Série **61**(2), 177–205 (1982)
49. Jockusch, C.G., Jr., McLaughlin, T.G.: Countable retracing functions and Π^0_2 predicates. Pacific Journal of Mathematics **30**, 67–93 (1969)
50. Jockusch, C.G., Jr., Soare, R.I.: Π^0_1 classes and degrees of theories. Transactions of the American Mathematical Society **173**, 33–56 (1972)
51. Kenny, R.: Effective zero-dimensionality for computable metric spaces. Logical Methods in Computer Science **11**, 1:11,25 (2015)
52. Kihara, T.: Effective forcing with Cantor manifolds, preprint. arXiv:1702.02630
53. Kihara, T.: Incomputability of simply connected planar continua. Computability **1**(2), 131–152 (2012)
54. Kihara, T.: Decomposing Borel functions using the Shore-Slaman join theorem. Fundamenta Mathematicae **230**, 1–13 (2015)
55. Kihara, T.: Borel-piecewise continuous reducibility for uniformization problems. Logical Methods in Computer Science **12**(4) (2016)
56. Kihara, T.: Higher randomness and lim-sup forcing within and beyond hyperarithmetic. In: Sets and Computations, *Lect. Notes Ser. Inst. Math. Sci. Natl. Univ. Singap.*, vol. 33, pp. 117–155. World Sci. Publ., Hackensack, NJ (2017)
57. Kihara, T., Pauly, A.: Point degree spectra of represented spaces. arXiv 1405.6866 (2014)
58. Kreisel, G., Lacombe, D.: Ensembles récursivement mesurables et ensembles récursivement ouverts et fermés. Comptes Rendus Académie des Sciences Paris **245**, 1106–1109 (1957)
59. Kushner, B.A.: Markov's constructive analysis; a participant's view. Theoretical Computer Science **219**, 267–285 (1999)
60. Le Roux, S., Pauly, A.: Finite choice, convex choice and finding roots. Logical Methods in Computer Science **11**(4), 4:6, 31 (2015)
61. Le Roux, S., Ziegler, M.: Singular coverings and non-uniform notions of closed set computability. Mathematical Logic Quarterly **54**(5), 545–560 (2008)
62. Louveau, A.: A separation theorem for Σ^1_1 sets. Transactions of the American Mathematical Society **260**(2), 363–378 (1980)
63. Manukjan, S.N.: Some topological peculiarities of constructive simple arcs. In: A. Markov, B.A. Kushner (eds.) Studies in the theory of algorithms and mathematical logic, Vol. 2 (Russian), pp. 122–129, 158. Vyčisl. Centr Akad. Nauk SSSR, Moscow (1976)
64. McNicholl, T.H.: The power of backtracking and the confinement of length. Proceedings of the American Mathematical Society **141**(3), 1041–1053 (2013)
65. McNicholl, T.H.: A note on the computable categoricity of ℓ^p spaces. In: A. Beckmann, V. Mitrana, M. Soskova (eds.) Evolving Computability, *Lecture Notes in Computer Science*, vol. 9136, pp. 268–275. Springer, Cham (2015), 11th Conference on Computability in Europe, CiE 2015, Bucharest, Romania, June 29–July 3, 2015
66. Melnikov, A.G.: Computably isometric spaces. The Journal of Symbolic Logic **78**(4), 1055–1085 (2013)
67. van Mill, J.: The Infinite-Dimensional Topology of Function Spaces, *North-Holland Mathematical Library*, vol. 64. North-Holland Publishing Co., Amsterdam (2001)
68. Miller, J.S.: Effectiveness for embedded spheres and balls. In: V. Brattka, M. Schröder, K. Weihrauch (eds.) CCA 2002 Computability and Complexity in Analysis, *Electronic Notes in Theoretical Computer Science*, vol. 66. Elsevier, Amsterdam (2002), 5th International Workshop, CCA 2002, Málaga, Spain, July 12–13, 2002
69. Miller, J.S.: Degrees of unsolvability of continuous functions. The Journal of Symbolic Logic **69**(2), 555–584 (2004)
70. Mori, T., Tsujii, Y., Yasugi, M.: Computability structures on metric spaces. In: D.S. Bridges, C.S. Calude, J. Gibbons, S. Reeves, I.H. Witten (eds.) Combinatorics, Complexity, and Logic, *Discrete Mathematics and Theoretical Computer Science*, pp. 351–362. Springer, Singapore (1997), proceedings of DMTCS'96
71. Moschovakis, Y.N.: Descriptive Set Theory, *Mathematical Surveys and Monographs*, vol. 155, second edn. American Mathematical Society, Providence, Rhode Island (2009)

72. Munkres, J.R.: Elements of Algebraic Topology. Addison-Wesley Publishing Company, Menlo Park, CA (1984)
73. Munkres, J.R.: Topology, 2nd edn. Prentice Hall, Upper Saddle River, NJ (2000)
74. Nadler, S.B., Jr.: Continuum Theory, *Monographs and Textbooks in Pure and Applied Mathematics*, vol. 158. Marcel Dekker, Inc., New York (1992)
75. Neumann, E.: Computational problems in metric fixed point theory and their Weihrauch degrees. Logical Methods in Computer Science **11**, 4:20,44 (2015)
76. Odifreddi, P.: Classical Recursion Theory - Volume II, *Studies in Logic and the Foundations of Mathematics*, vol. 143. North-Holland, Amsterdam (1999)
77. van Oosten, J.: Realizability: An Introduction to Its Categorical Side, *Studies in Logic and the Foundations of Mathematics*, vol. 152. Elsevier B. V., Amsterdam (2008)
78. Pauly, A.: On the topological aspects of the theory of represented spaces. Computability **5**(2), 159–180 (2016)
79. Pauly, A., de Brecht, M.: Non-deterministic computation and the Jayne-Rogers theorem. In: B. Löwe, G. Winskel (eds.) Proceedings 8th International Workshop on Developments in Computational Models, DCM 2012, Cambridge, United Kingdom, 17 June 2012., *Electronic Proceedings in Theoretical Computer Science*, vol. 143, pp. 87–96 (2014)
80. Pauly, A., de Brecht, M.: Descriptive set theory in the category of represented spaces. In: 30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), pp. 438–449 (2015)
81. Porter, M., Day, A., Downey, R.: Notes on computable analysis. Theory of Computing Systems **60**(1), 53–111 (2017)
82. Pour-El, M.B., Richards, J.I.: Computability in Analysis and Physics. *Perspectives in Mathematical Logic*. Springer, Berlin (1989)
83. Schröder, M.: Admissible representations for continuous computations. Ph.D. thesis. Fachbereich Informatik, FernUniversität Hagen (2002)
84. Schröder, M.: Extended admissibility. Theoretical Computer Science **284**(2), 519–538 (2002)
85. Shastri, A.R.: Elements of Differential Topology. CRC Press, Boca Raton, FL (2011)
86. Simonsen, J.G.: On local non-compactness in recursive mathematics. Mathematical Logic Quarterly **52**(4), 323–330 (2006)
87. Simpson, S.G.: Subsystems of Second Order Arithmetic, 2nd edn. *Perspectives in Logic*. Cambridge University Press (2009)
88. Simpson, S.G.: Mass problems associated with effectively closed sets. The Tohoku Mathematical Journal. Second Series **63**(4), 489–517 (2011)
89. Simpson, S.G.: Implicit definability in arithmetic. Notre Dame Journal of Formal Logic **57**(3), 329–339 (2016)
90. Solovay, R.M.: On the cardinality of $\sum^1_2$ sets of reals. In: J.J. Bulloff, T.C. Holyoke, S.W. Hahn (eds.) Foundations of Mathematics, pp. 58–73. Springer, New York (1969), Symposium Commemorating Kurt Gödel, Columbus, Ohio, 1966
91. Solovay, R.M.: A model of set-theory in which every set of reals is Lebesgue measurable. Annals of Mathematics. Second Series **92**, 1–56 (1970)
92. Taylor, P.: A lambda calculus for real analysis. Journal of Logic and Analysis **2**, Paper 5, 115 (2010)
93. Taylor, P.: Foundations *for* computable topology. In: G. Sommaruga (ed.) Foundational Theories of Classical and Constructive Mathematics, *Western Ontario Series in Philosophy of Science*, vol. 76, pp. 265–310. Springer, Dordrecht (2011)
94. Weihrauch, K.: Computability on computable metric spaces. Theoretical Computer Science **113**, 191–210 (1993)
95. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)
96. Weihrauch, K.: Computable separation in topology, from T_0 to T_2. Journal of Universal Computer Science **16**(18), 2733–2753 (2010)
97. Weihrauch, K.: Computable planar curves intersect in a computable point. Computability **8**(3, 4), 399–415 (2019)

98. Weihrauch, K., Grubba, T.: Elementary computable topology. Journal of Universal Computer Science **15**(6), 1381–1422 (2009)
99. Weihrauch, K., Tavana-Roshandel, N.: Representations of measurable sets in computable measure theory. Logical Methods in Computer Science **10**, 3:7,21 (2014)
100. Wu, Y., Ding, D.: Computability of measurable sets via effective metrics. Mathematical Logic Quarterly **51**(6), 543–559 (2005)
101. Wu, Y., Ding, D.: Computability of measurable sets via effective topologies. Archive for Mathematical Logic **45**(3), 365–379 (2006)
102. Yasugi, M., Mori, T., Tsujii, Y.: Effective properties of sets and functions in metric spaces with computability structure. Theoretical Computer Science **219**, 467–486 (1999)
103. Ziegler, M.: Computability on regular subsets of Euclidean space. Mathematical Logic Quarterly **48**(Suppl. 1), 157–181 (2002)
104. Ziegler, M.: Computable operators on regular sets. Mathematical Logic Quarterly **50**(4,5), 392–404 (2004)

Chapter 3
Computability of Differential Equations

Daniel S. Graça and Ning Zhong

Abstract In this chapter, we provide a survey of results concerning the computability and computational complexity of differential equations. In particular, we study the conditions which ensure computability of the solution to an initial value problem for an ordinary differential equation (ODE) and analyze the computational complexity of a computable solution. We also present computability results concerning the asymptotic behaviors of ODEs as well as several classically important partial differential equations.

3.1 Introduction

A differential equation is an equation that relates a function with its (partial) derivatives of various orders. Differential equations are widely used in many fields and have been extensively studied since the time when the calculus was invented. Given a differential equation, the goal is to find its solutions. Unfortunately, most differential equations do not possess solutions in explicit forms. In order to obtain useful information about the solutions without having an explicit representation formula for them, various methods and tools have been developed. Classically, the following methods are of most significance:

- *Numerical methods*: where solutions are approximated numerically.
- *Qualitative methods*: where the behavior of a differential equation and of its solutions is analyzed from a qualitative perspective.

Daniel S. Graça
Universidade do Algarve, Faro, Portugal and Instituto de Telecomunicações, Portugal
e-mail: dgraca@ualg.pt

Ning Zhong
Department of Mathematical Sciences, University of Cincinnati, Cincinnati, Ohio, USA
e-mail: ning.zhong@uc.edu

V. Brattka, P. Hertling (eds.), *Handbook of Computability and Complexity in Analysis*,
Theory and Applications of Computability, https://doi.org/10.1007/978-3-030-59234-9_3

Numerical methods generate numerical algorithms which calculate approximations to the exact solution that cannot be found explicitly, and the approximations are expected to converge to the exact solution. The convergence is a fundamental property for a numerical algorithm to be of any use. In numerical analysis, the accuracy of approximation (the convergence rate) is usually not specified – sometimes it is impossible to specify (see [81] for an example) – as an input to the algorithm that computes the approximations, but rather it is provided externally and implicitly by finding certain analytical forms of approximation errors, which frequently depend on additional assumptions; for example, the twice differentiability of the exact solution. This requirement is often crucial, yet not applicable to many initial value problems for the first-order ODEs whose solutions may not be C^2 but nevertheless are approximable by convergent numerical solutions.

The qualitative method is another key tool, often complementing numerical methods, in the study of ordinary differential equations; in particular, in describing the asymptotic behaviors of solutions of ODEs. For example, it is used to describe the limit set of a typical solution as a spatial object as time tends to infinity.

In addition to the classical methods, there is also a relatively new approach based on the Turing machine model over the real field (see [67, 56, 87] and references therein):

- *Computable analysis methods*: where computability and computational complexity of problems related to differential equations are studied.

In computable analysis, a problem is called computable if it can in principle be solved or approximated with arbitrary precision by a computing device. The computable analysis method provides useful insights on many problems related to a given ODE from the perspective of computation in the sense that (1) the method can be used to identify whether a problem is computable; (2) if a problem is proved to be computable using the computable analysis method, often the proof generates a (Turing) algorithm that computes approximations with arbitrary precision and, moreover, with the accuracy specified as a part of the input (computability); and (3) the method has a mechanism for assessing the resources needed for a computation (computational complexity).

The computable analysis method has been used by many authors to explore the computational aspects of differential equations; in particular, to address questions related to computing solutions of differential equations. In this chapter, we present a brief survey of some existing results.

The chapter is organized as follows. Section 3.2 addresses computability of solutions of first-order ODEs, while Section 3.3 analyzes the computational complexity of the computable solutions. Section 3.4 discusses computability of asymptotic behaviors of ODEs. Finally, Section 3.5 presents some computability results of several classically important partial differential equations.

3.2 Computability of the Solutions of Ordinary Differential Equations

In this section, we consider ordinary differential equations (ODEs) of the form

$$y' = f(y) \tag{3.1}$$

and associated initial value problems (IVPs)

$$\begin{cases} y' = f(y) \\ y(t_0) = y_0 \end{cases} \tag{3.2}$$

where $f : E \to \mathbb{R}^n$ is continuous over $E \subseteq \mathbb{R}^n$. Since an ODE of the more general form $y' = g(t,y)$ can be reduced to (3.1) if one replaces the independent variable t by a new variable y_{n+1} satisfying $y'_{n+1} = 1$, $y_{n+1}(t_0) = t_0$, there is no loss of generality in studying only ODEs in the form of (3.1).

3.2.1 Computability over Compact Sets

Assume that E is a compact subset of $\mathbb{R}^n$. Classically, it is well known that continuity of f ensures the existence of solution(s) to the IVP (3.2) (Peano's existence theorem, see, e.g., [11, pp. 191-193]), and if f is Lipschitz continuous then the solution is unique and C^1 smooth (Picard-Lindelöf theorem, see, e.g., [43, pp. 8-10]). The compactness of E is essential for the proof of Peano's existence theorem, in which a sequence of functions is constructed and then the Arzelà-Ascoli theorem is employed to obtain a subsequence that converges toward a solution of (3.2). The proof is non-constructive. On the other hand, the classical proof of existence and uniqueness for Lipschitz continuous f is constructive, such as Picard's scheme of successive approximations; such constructive schemes lead naturally to some computer programs for computing the unique solution, uniformly in f. Recall that f is said to be Lipschitz continuous if it satisfies a *Lipschitz condition* on E; i.e., there exists some positive constant K (a *Lipschitz constant*) such that, for all $x, y \in E$

$$\|f(x) - f(y)\| \leq K \|x - y\| .$$

Thus the question remaining to be investigated is whether solutions are still computable when the vector field f is continuous on E but fails to be Lipschitz continuous. In this case, whether the IVP (3.2) has a unique solution plays a pivotal role in determining whether (3.2) admits a computable solution. A theorem by Osgood [10] shows implicitly that the initial value problem $y' = f(t,y)$, $y(0) = y_0$ does admit a computable solution on $[0,\delta]$ for some $\delta > 0$, provided that f is computable on $E = [0,1] \times [-1,1]$ and the solution of the IVP is unique. The idea used for the proof is to compute sequences of functions which converge to a maximal solution y_{max} from above and a minimal solution y_{min} from below. If the solution is unique,

then the convergence is effective and thus results in a computable solution. However, this technique only works in the plane, since in higher dimensions there is no similar notion of a maximal or a minimal solution. In [71] Ruohonen generalizes this result to higher dimensions, using approximation funnels to compute the solution of (3.2), obtaining the following result.

Theorem 3.2.1. *Suppose that f is computable and not identically zero on the closed ball $B(y_0,b) = \{x \in \mathbb{R}^n : |y_0 - x| \leq b\}$, where $b > 0, t_0, y_0$ are computable. Then the solution of (3.2) is computable on $[t_0, t_0 + \delta]$ for $\delta = \frac{b}{max_{x\in E}\|f(x)\|} > 0$, provided that the solution of (3.2) is unique on that interval.*

Another result that relies on Picard's construction to compute the solution of (3.2) is presented in [28]. There a domain-theoretic version of Picard's operator is presented to compute the solution of the IVP (3.2), provided that the function defining the IVP is Lipschitz continuous.

It is also shown in [19] that, under certain conditions, solutions of the class of implicit ODEs in the form of $A(y) \cdot y' = f(y, \lambda)$ are computable, where $A(y)$ is an $n \times n$ matrix formed by polynomials in $y_1, \ldots, y_n$, with $y = (y_1, \ldots, y_n)$, and λ is a parameter.

The scenario changes completely if (3.2) admits multiple solutions. In this case, it is possible that none of the solutions is computable. The following result was first proved by Aberth [2, p. 152] for the case where f is assumed just to be computable; later a different proof was presented by Pour-El and Richards [65]; then the result was extended in [55] to include the case where f is polynomial-time computable.

Theorem 3.2.2. *There exists a polynomial-time computable function $f : [0,1] \times [-1,1] \to \mathbb{R}$ such that*

$$\begin{cases} y' = f(t,y) \\ y(0) = 0 \end{cases} \tag{3.3}$$

does not have a computable solution y on $[0,\delta]$ for any $\delta > 0$.

The idea for the proof is to construct a function f that embeds a non-computable problem $\mathscr{P}$ and show that if a computable solution for (3.3) exists, then $\mathscr{P}$ can be solved computationally, a contradiction. For details of the proof, we refer the reader to the original papers [65, 55] or to [56, pp. 216-219].

3.2.2 Computability over Non-compact Sets

In this subsection, we assume that E is an open subset of $\mathbb{R}^n$. The Peano existence theorem indicates that the IVP (3.2) has a solution existing on some time interval containing t_0 in its interior, provided that f is continuous on E. An interval is called an interval of existence if it contains t_0 and if a solution of (3.2) exists on it. The following classical result indicates how large an interval of existence may be (see, e.g., [43, pp. 12-13]).

Proposition 3.2.3. *(Maximal Interval of Existence) Let $f : E \to \mathbb{R}^n$ be continuous on E. Then the IVP (3.2) has a maximal interval of existence, and it is of the form (α,β), with $\alpha \in [-\infty,\infty)$ and $\beta \in (-\infty,\infty]$. There is a solution $y(t)$ of (3.2) on (α,β) and $y(t)$ leaves every compact subset of E, i.e., $y(t)$ tends to the boundary ∂E of E, as $t \to \alpha$ from above and as $t \to \beta$ from below.*

The proposition is a simple classical existence statement "the IVP (3.2) has a maximal interval of existence and a solution exists on it." It turns out that this simple statement gives rise to a variety of questions about computability. Is the maximal interval (α,β) of existence computable from the data (f,t_0,y_0) that defines the IVP (3.2)? Does (3.2) admit a computable solution on (α,β)? If $E = \mathbb{R}^n$, then a solution of (3.2) is said to blow up in finite time if α and β are two real numbers. Is the problem whether (3.2) admits a finite blowup solution decidable? We will discuss these questions in this subsection.

It is known from Subsection 3.2.1 that if the IVP (3.2) has multiple solutions then it may occur that none of the solutions is computable. Hence, in this subsection, we consider only those IVPs to which the solution is unique.

We begin by examining the proof of the proposition. The proposition can be proved by using the Peano existence theorem repeatedly: The theorem first produces a solution $y(t)$ of the IVP (3.2) on an interval $[a_0,b_0]$ containing t_0; then it extends $y(t)$ to $[a_1,a_0]$ and to $[b_0,b_1]$, $a_1 < a_0$ and $b_1 > b_0$, by yielding a solution to the IVP $v' = f(v)$, $v(b_0) = y(b_0)$, and to the IVP $v' = f(v)$, $v(a_0) = y(a_0)$; and the process is repeated infinitely many times to reach the maximality: $(\alpha,\beta) = \bigcup_{j+1}^{\infty}[a_j,b_j]$. Thus, if the IVP (3.2) has a unique solution $y(t)$ over (α,β), then it follows from Theorem 3.2.1 that a_j, b_j, and $y(t)$ on interval $[a_j,b_j]$ are computable from j, f, t_0, and y_0. But the argument is insufficient to show that $y(t)$ is computable on (α,β) because it is possible that there does not exist a "master" algorithm computing $y(t)$, independent of j. One possible way of dealing with this shortcoming is to strengthen f to be a smoother function, which is actually an approach frequently used in classical analysis of ODEs: the smoother the vector field f is, the better, possibly, the solution y behaves. This is the approach used in [36]. By strengthening f to be a C^1 function on E, the following result is shown in [36].

Theorem 3.2.4. *Assume that E is an r.e. open subset of $\mathbb{R}^n$. Consider the initial value problem (3.2) where f is C^1 on E. Let (α,β) be the maximal interval of existence of the solution $y(\cdot)$ of (3.2) on E. Then:*

1. *The operator $(f,t_0,y_0) \mapsto y(\cdot)$ is computable;*
2. *The operator $(f,t_0,y_0) \mapsto (\alpha,\beta)$ is semi-computable (i.e., α can be computed from above and β can be computed from below).*

Recall that E is an r.e. open subset of $\mathbb{R}^n$ if it can be filled up by a computer-generated sequence $\{B_j\}_{j=1}^{\infty}$ of rational balls (balls having rational centers and rational radii), i.e., $E = \bigcup_{j=1}^{\infty} B_j$. The main role played by the C^1 smoothness in the proof is that C^1 smoothness implies local Lipschitz continuity (and the converse is false); that is, if f is C^1 on E, then f is Lipschitz continuous on every compact subset of E. An algorithmic version of this fact is proved in [36]. Thus, when giving a

C^1-name of f, it becomes possible to compute a Lipschitz constant K_j for each E_j, where E_j is the closure of $\bigcup_{i=1}^{j} B_i$, then compute the solution of (3.2) on E_j, and thus compute the solution over the maximal interval (α,β) of existence, because the E_j are getting ever larger and eventually fill up E.

Although for many important ODEs their vector fields f are indeed C^1, the requirement for f being C^1 is nevertheless not necessary for ensuring the existence of a unique solution. It is then natural to ask whether it is possible to compute the unique solution of the IVP (3.2), whenever it exists, over the maximal interval (α,β); in other words, whether Theorem 3.2.4 holds true over (α,β) when E is an open subset of $\mathbb{R}^n$. The problem is studied in [26] and a positive answer is presented there. The proof uses a quite different approach; the idea underlying the approach is to try to cover the solution with rational boxes and to test (i) and (ii), in an algorithmic way: (i) whether a given set of rational boxes is an actual covering of the solution of (3.2), and (ii) whether the "diameter of the covering" is sufficiently small so that the rational boxes provide an approximation of the solution with the desired accuracy whenever (i) is satisfied. The process of enumerating all possible families of rational boxes and applying the tests (i) and (ii) to each family generates, effectively, better and better approximations to the unique solution of (3.2), and thus proves that the solution is computable. This procedure can also be applied to study the computability of differential inclusions (see [26] for more details). The following theorem proved in [26] (see also [25]) strengthens Theorem 3.2.4.

Theorem 3.2.5. *Consider the initial value problem (3.2) where f is continuous on the open set E. Suppose that (3.2) has a unique solution $y(\cdot)$ on E, defined on the maximal interval of existence (α,β). Then*

1. *The operator $(f,t_0,y_0) \mapsto y(\cdot)$ is computable;*
2. *The operator $(f,t_0,y_0) \mapsto (\alpha,\beta)$ is semi-computable (i.e., α can be computed from above and β can be computed from below).*

In particular, if f is a computable function and t_0, y_0 are computable points, then (α,β) is a recursively enumerable open set and the solution $y(\cdot)$ is a computable function.

A question that remains is whether the maximal interval of existence is computable. The answer is negative, as shown in [36].

Theorem 3.2.6. *There is an analytic and computable function $f : \mathbb{R} \to \mathbb{R}$ such that the unique solution of the problem*

$$\begin{cases} y' = f(y) \\ y(0) = 0 \end{cases} \tag{3.4}$$

is defined on a non-computable maximal interval of existence.

The theorem indicates that the computability of the maximal interval (α,β) of existence is not tied up with the smoothness of f; rather it has more to do with

the "globalness" of (α, β). Later, in Section 3.4, we will see again that a "global property" is usually difficult to compute. The counterexample is constructed by creating a computable odd and bijective function $\varphi : (-\alpha, \alpha) \to \mathbb{R}$, where α is a non-computable real number, such that φ satisfies the following conditions: (i) φ is the solution of (3.4) for an analytic and computable function f; and (ii) $\varphi(x) \to \pm\infty$ as $x \to \pm\alpha^{\mp}$.

When E is the whole space $\mathbb{R}^n$ and f is defined on $\mathbb{R}^n$, if a solution $y(t)$ of (3.2) is defined for all $t \geq t_0$, it is called a (positively) global solution that does not blow up in finite time. In general, it is difficult to predict whether or not a solution will blow up in finite time from a given initial datum (t_0, y_0), as Theorem 3.2.6 already indicates, because it often requires extra knowledge on some quantitative estimates and asymptotics of the solution over a long period of time. Actually the problem of determining whether the solution of (3.2) blows up in finite time is undecidable even if $f : \mathbb{R}^n \to \mathbb{R}^n$ has only polynomials as components [39]. Nevertheless, certain computational insight on the blowup problem is obtainable from f, as the following theorem in [69] suggests.

Theorem 3.2.7. *Consider the IVP (3.2), where f is locally Lipschitz. Let $\mathscr{Z}$ be the set of all initial values (t_0, y_0) for which the corresponding unique solution of (3.2) is (positively) global. Then $\mathscr{Z}$ is a G_δ-set and there is a computable operator determining $\mathscr{Z}$ from f.*

3.3 Computational Complexity of the Solutions of Ordinary Differential Equations

In this section we analyze the computational complexity of solving an initial value problem (3.2).

We begin by noting that, as mentioned before, any C^1 function defined on a compact set $E \subseteq \mathbb{R}^n$ satisfies a Lipschitz condition. The analysis of any standard algorithm (e.g., Euler's method, or the algorithm implicit in Picard's iteration method) to solve an initial value problem (3.2) applied to a continuous function which satisfies a Lipschitz condition over a compact set, with a Lipschitz constant $K > 0$, shows that the value of the Lipschitz constant affects the number of iterations needed to compute the solution of (3.2) with a given accuracy. However, since the Lipschitz constant is constant in a compact set, it can be hidden in the big O notation. When we solve (3.2) over a non-compact set, we typically have infinitely many distinct local Lipschitz constants, and the complexity results need to take into account this effect. This is the reason the two cases are analyzed in separate sections, as for the case of computability.

3.3.1 Results for Compact Sets

We have just discussed above the importance that a Lipschitz condition seems to have when computing the solution of an initial value problem (3.2). First it makes sense to ask how the presence of a Lipschitz constant affects the computational complexity of the solution of (3.2). We already know from Theorem 3.2.1 that if (3.2) has a unique solution, then it must be computable. So what is the complexity of the solution of (3.2) if we assume that f is "easy to compute" (polynomial-time computable)? The following result from [56, p. 219], based on another result of Miller [58, p. 469], shows that in that case the solution of (3.2), while computable, can have arbitrarily high complexity.

Theorem 3.3.1. *Let a be an arbitrary computable real number in $[0,1]$. Then there is a polynomial-time computable function f defined on $[0,1]\times[-1,1]$ such that $y(t)=at^2$ is the unique solution of the initial value problem defined by*

$$y' = f(t,y) \text{ and } y(0)=0. \tag{3.5}$$

Note that in (3.5) we have used the independent variable t explicitly, i.e., we considered a non-autonomous ODE instead of an autonomous ODE as in (3.2). Technically speaking, that is not needed as we have seen in Section 3.2, since a non-autonomous IVP can be converted into an autonomous IVP. However, in the remainder of this section we will explicitly consider the independent variable t since, as we will see later in this section, this will allow us to do a more refined analysis on what happens at a computational complexity level by analyzing how the smoothness of f relative to t or to y can affect the complexity of the solution of (3.5). In this section we also suppose that the initial condition for (3.5) is $y(0)=0$, although WLOG any initial condition $y(t_0)=y_0$, where t_0, y_0 are polynomial-time computable, could be used to obtain the same results. We note that for non-autonomous IVPs (3.5) the Lipschitz condition is not needed for the variable t. Therefore, in that case, we say that $f:[0,1]\times B(0,1)\to\mathbb{R}^n$, with $B(0,1)\subseteq\mathbb{R}^n$, is (right-)Lipschitz if it satisfies the following condition

$$\|f(t,y_1)-f(t,y_2)\| \le L\|y_1-y_2\|$$

for all $t\in[0,1]$ and all $y_1,y_2\in B(0,1)\subseteq\mathbb{R}^n$. Remark that if f in (3.5) is (right-) Lipschitz, then when we convert the ODE of the initial value problem (3.5) into an autonomous ODE (3.1), then f in (3.1) is Lipschitz continuous. Therefore, using non-autonomous ODEs with (right-)Lipschitz ODEs or autonomous Lipschitz continuous ODEs are equivalent approaches, although the non-autonomous case (3.5) has the advantage mentioned above.

In (3.5) we also assume that the function f is defined on $[0,1]\times B(0,1)$. However, those results can easily be extended to functions defined over $[0,T]\times B(0,M)$ for $T,M>0$ via rescaling. More concretely, let $f:[0,T]\times B(0,M)\to\mathbb{R}^n$ and let $y:[0,T]\to B(0,M)$ be the solution of (3.5). Note that we suppose that $y(t)$ is always defined for all $t\in[0,T]$ (and hence is always inside $B(0,M)$). Define the function

$y_* : [0,1] \to \mathbb{R}^n$ as $y_*(t) = y(tT)$. In other words $y_*(1) = y(T)$. It is not difficult to see that y_* is the solution of the IVP

$$y'_* = Tf(tT, y_*) \text{ and } y_*(0) = 0.$$

Similarly one can define the function $y_{**} : [0,1] \to \mathbb{R}^n$ as $y_{**}(t) = y_*(t)/M = y(tT)/M$. In other words $y_{**}(t)$ is just like y_*, but scaled down by a factor of M so that $y_{**}(t) \in B(0,1)$ for all $t \in [0,1]$. It is also not difficult to check that y_{**} is the solution of the IVP

$$y'_{**} = \frac{T}{M} f(tT, My_{**}) \text{ and } y_{**}(0) = 0.$$

Note that, for a fixed compact set $[0,T] \times B(0,M)$, T and M are just constants. Therefore the complexity of $f : [0,T] \times B(0,M) \to \mathbb{R}^n$ is essentially the same as that of $f_{**} : [0,1] \times B(0,1) \to \mathbb{R}^n$ given by $f_{**}(t,y) = \frac{T}{M} f(tT, My)$ and thus the complexity of finding the solution of (3.5) when $f : [0,T] \times B(0,M) \to \mathbb{R}^n$ is essentially the same as that of finding the solution of (3.5) when $f : [0,1] \times B(0,1) \to \mathbb{R}^n$.

In [23, p. 450] it is shown that if f of (3.5) satisfies a Lipschitz condition on $[0,1] \times [-1,1]$ and f belongs to the nth level of the Grzegorczyk hierarchy $\mathscr{E}^{(n)}$, for $n \geq 3$, then so will its solution y. This result is refined for the case of polynomial-time computable functions f in [55, p. 159] (or [56, p. 221]). There it is mentioned that if f of (3.5) is polynomial-time computable and satisfies a Lipschitz condition on $[0,1] \times [-1,1]$, then the solution of (3.4) can be computed in polynomial space. This result can be extended to the case where $y : [0,1] \to B(0,1)$ where $B(0,1) \subseteq \mathbb{R}^n$ and $n \geq 1$.

Theorem 3.3.2. *Let $f : [0,1] \times B(0,1) \to \mathbb{R}^n$, with $B(0,1) \subseteq \mathbb{R}^n$, be a polynomial-time computable (right-)Lipschitz function and assume that $y : [0,1] \to B(0,1)$ is the solution of (3.5). Then y is polynomial-space computable.*

As mentioned by Ko, the proof of this result follows from a careful analysis of Euler's method. For example, it is shown in [7, pp. 349-350] that when $n = 1$ and we use Euler's method to compute the solution y of (3.4) over the time interval $[0,1]$ with time step bounded by $h > 0$ and rounding error bounded by $\rho > 0$, the error e_n in step n is bounded by

$$|e_n| \leq A\rho + B(hC + \rho/h) \tag{3.6}$$

where $A, B, C > 0$ are some constants (the original formula (6.2.32) [7] explicitly tells how A, B, C depend on the Lipschitz constant, etc., but that level of detail is not needed for our analysis). So to compute the solution of (3.4) over the time interval $[0,1]$ with accuracy 2^{-n} we could use Euler's method with $h = a2^{-n}$ and $\rho = b(2^{-n})^2$ for some appropriate constants $a, b > 0$ which are independent of n. Euler's method would then run in space polynomial (indeed quadratic) in n, which shows the result. The case $n > 1$ can be obtained by considering the Euler method for dimensions $n > 1$ as explained in [11, Section 7.2]. The analysis of the error (done in [11, Section 7.3], except that the rounding error is not considered, but this can also be taken into account using a procedure similar to that of [7, p. 350]) gives a

formula (3.6), but with constants $A,B,C > 0$ which have different values than from the previous case, and therefore the conclusions remain valid for the case where $n > 1$.

In [56, Section 7.4] (also implicitly in [55, p. 159, Question D],) Ko asks whether polynomial space is also a lower bound for the complexity of the solution y of (3.5), assuming the conditions of the previous theorem for $n = 1$. Kawamura provides an affirmative answer in [49] (see also [48]) using an appropriate notion of reduction (see [49, Section 2.2]) which allows us to define $\mathscr{C}$-hard and $\mathscr{C}$-complete functions for a complexity class $\mathscr{C}$. Namely Kawamura shows the following theorem.

Theorem 3.3.3. *There exists a polynomial-time computable function $f : [0,1] \times [-1,1] \to \mathbb{R}$ which satisfies a (right-)Lipschitz condition, such that the unique solution $y : [0,1] \to \mathbb{R}$ of (3.5) takes values in $[-1,1]$ and is PSPACE-complete.*

As a corollary [49, Corollary 3.3] one has the following result.

Corollary 3.3.4. *$P = PSPACE$ if and only if any solution $y : [0,1] \to B(0,1)$ of (3.5) is polynomial-time computable, where $f : [0,1] \times B(0,1) \to \mathbb{R}^n$, with $B(0,1) \subseteq \mathbb{R}^n$, is a polynomial-time (right-)Lipschitz computable function.*

The previous results are non-uniform since we fix the complexity of the input to be polynomial time. But one can naturally ask what is the (uniform) complexity of the operator $LipIVP$ which maps a function f satisfying a Lipschitz condition to the solution y of (3.5). A major problem to tackle is that until recently there was no general theory to measure the complexity of operators which map functions into functions. This problem was solved in [50] using appropriate representations of functions and appropriate notions of reduction, hardness and completeness. Using this setting one can define $\mathscr{C}$-hard and $\mathscr{C}$-complete functions for a complexity class $\mathscr{C}$. Since we are talking of functions, it makes sense to take $\mathscr{C} = FP$ (functions computable in polynomial time) or $\mathscr{C} = FPSPACE$ (functions computable in polynomial space). By using appropriate representations $\delta_\square$ for continuous functions in $C([0,1])$ and $\delta_{\square L}$ for functions satisfying a Lipschitz condition (which are continuous), and an appropriate reduction $\leq^2_{mF}$(see [50] for more details) it was also shown in [50, Theorem 4.10] that the operator $LipIVP$ is FPSPACE-complete.

Theorem 3.3.5. *$LipIVP$ is $(\delta_{\square L}, \delta_\square)$-$FPSPACE$-$\leq^2_{mF}$-complete.*

An interesting question is whether smoothness of f helps reduce the computational complexity of solving an IVP (3.5). We say that $f : [0,1] \times \mathbb{R} \to \mathbb{R}$ is of class $C^{(i,j)}$ if the partial derivative $\partial^{n+m} f(t,y)/\partial t^n \partial y^m$ exists and is continuous for all $n \leq i$ and $m \leq j$; it is said to be of class $C^{(\infty,j)}$ if it is of class $C^{(i,j)}$ for all $i \in \mathbb{N}$. In [51] this question is analyzed and Theorem 3.3.3 is generalized to $C^{(\infty,1)}$ functions.

Theorem 3.3.6. *There exists a polynomial-time computable function $f : [0,1] \times [-1,1] \to \mathbb{R}$ of class $C^{(\infty,1)}$ such that the unique solution $y : [0,1] \to \mathbb{R}$ of (3.5) is PSPACE-complete.*

In that paper it is also shown [51, Theorem 2] that if f is more than once differentiable, then the unique solution can be CH-hard, where $CH \subseteq PSPACE$ is the counting hierarchy.

Theorem 3.3.7. *Let k be a positive integer. There exists a polynomial-time computable function $f : [0,1] \times [-1,1] \to \mathbb{R}$ of class $C^{(\infty,k)}$ such that the unique solution $y : [0,1] \to \mathbb{R}$ of (3.5) is CH-hard.*

In the most extreme case, when f is analytic, the solution of (3.5) is polynomial-time computable on a compact set. This follows from results of Müller [60] and Ko and Friedman [57], which show that polynomial-time computability of an analytic function on a compact interval is equivalent to polynomial-time computability of the sequence of its Taylor coefficients at a rational point. Given a sequence of Taylor coefficients for f one can compute the Taylor coefficients for the solution y of (3.5) [61, Theorem 2.1]. Note that it is known that if f is analytic, then so is the solution y of (3.5) [5, Section 32.4]. By analytic continuation [60], polynomial-time computability of y follows.

Theorem 3.3.8. *Let $f : [0,1] \times B(0,1) \to B(0,1)$ with $B(0,1) \subseteq \mathbb{R}^n$ be a polynomial-time computable analytic function and $y : [0,1] \to B(0,1)$ be the solution of (3.5). Then y is polynomial-time computable.*

As remarked in [51, last paragraph of Section 5.2], the previous argument also shows uniform polynomial-time computability of the operator $LipIVP$, if we represent analytic functions by their sequence of Taylor coefficients, obtaining a representation δ_{Taylor}, since the Taylor sequence of y is easy to compute from the Taylor sequence of f. In the next theorem $LipIVP\restriction^{\mathscr{D}}$ represents the operator $LipIVP$ restricted to the class of (real) functions $\mathscr{D}$ and C^ω represents the class of analytic functions.

Theorem 3.3.9. *The operator $LipIVP\restriction^{C^\omega}$ belongs to $(\delta_{Taylor}, \delta_{Taylor})$-$FP$.*

Theorem 3.3.6 can also be extended to a uniform version by showing that the operator $LipIVP\restriction^{C^{(\infty,1)}}$ [51, last paragraph of Section 5.2] is $FPSPACE$ complete, using the polynomial-time Weihrauch reduction $\leq_W$.

Theorem 3.3.10. *The operator $LipIVP\restriction^{C^{(\infty,1)}}$ is $(\delta_{\square L}, \delta_{\square})$-$FPSPACE$-$\leq_W$-complete.*

Computability in polynomial time of the solution of (3.5) is also obtained in [29] by using polynomial enclosures in Picard's method, provided that some assumptions are made on the function f in (3.5), such as that f is Lipschitz and that its representation via polynomial enclosures satisfies certain properties (e.g., this representation should not take "too much space").

3.3.2 Results for Non-compact Sets

In the previous section we have analyzed the computational complexity of solutions of an IVP (3.5) (or, equivalently, (3.2)) where f is considered over a compact set $[0,1] \times B(0,1)$, with $B(0,1) \subseteq \mathbb{R}^n$. But what happens when $f : \mathbb{R}^{n+1} \to \mathbb{R}^n$ and the solution y of (3.5) is considered over its maximal interval of definition? That is, what is the computational complexity of $y : \mathbb{R} \to \mathbb{R}^n$, or more generally of the operator mapping f to y?

As a first attempt we could consider the approach, given $t \in [0,+\infty)$, of rescaling the IVP (3.5) given by $f : [0,t] \times B(0,M) \to \mathbb{R}^n$, where M is big enough so that $y(t) \in B(0,M)$ for all $t \in [0,t]$, to an IVP (3.5) which uses a function $\bar{f} : [0,1] \times B(0,1)$ instead of f, as explained in the previous section. Since, for a fixed M (and fixed T, with $t \leq T$), the complexity of solving (3.5) is the same as the complexity of solving (3.5) where $\bar{f}$ is used instead of f, we might be led to conclude that the complexity of obtaining the solution of (3.5) for the non-compact case where $f : \mathbb{R}^{n+1} \to \mathbb{R}^n$ is the same as when f is taken over a compact set $[0,1] \times B(0,1)$. However this reasoning is incorrect, since M (and T) is not fixed in this case and must be uniformly computed from y, i.e., $M = M(y(t)) = M(t)$. This is not a trivial task since to compute $y(t)$, we need to know $M(t)$. On the other side, to know $M(t)$, we need to know $\max_{0 \leq u \leq t} \|y(u)\|$ and therefore we get into a circular argument where $M(t)$ is needed to compute $y(t)$ and vice versa. To better illustrate this problem, consider the following system taken from [64]

$$\begin{cases} y_1(0) = 1 \\ y_2(0) = 1 \\ \quad \cdots \\ y_d(0) = 1 \end{cases} \qquad \begin{cases} y_1'(t) = y_1(t) \\ y_2'(t) = y_1(t) y_2(t) \\ \quad \cdots \\ y_d'(t) = y_1(t) \cdots y_d(t) \end{cases}$$

The results of Section 3.3.1 show that for any fixed, compact $I = [0,T]$ the solution y is polynomial-time computable since the ODE is analytic. On the other hand, this system can be solved explicitly and yields:

$$y_1(t) = e^t \qquad y_{i+1}(t) = e^{y_i(t)-1} \qquad y_d(t) = e^{e^{\cdots^{e^{e^t}-1}}-1}$$

One immediately sees that y_d being a tower of exponentials prevents y from being polynomial-time computable over $\mathbb{R}$, since one might need (supra-)exponential time just to write down an integer approximation of y_d with precision $1/2 = 2^{-1}$ for $d \geq 2$. This example shows that the solution of an analytic IVP (or even of a polynomial IVP) can be polynomial-time computable on any fixed compact set, while it may not necessarily be polynomial-time computable over $\mathbb{R}$.

A possible way to solve this problem is to analyze the complexity of (3.5) using a bound on the growth of the solution y as a parameter on the function used to measure the complexity (this is an example of *parametrized complexity*), since the problem of knowing how quickly y can grow is not generally well understood, even

when f is constituted by polynomials. This approach is taken in [13] for the case of polynomial IVPs (solutions of polynomial IVPs (3.5) where f is formed by polynomials are sometimes called PIVP functions) and in [12] for the case of analytic functions. The motivation for studying PIVPs is that the class of PIVP functions is well behaved and includes many interesting functions. In particular it contains all of the usual functions of analysis (exponential, trigonometric functions, polynomials, their inverses) and is closed under the usual arithmetic operations, composition, and ODE solving [21, 34]. It can also be shown that this class is exactly the class of functions generated by Shannon's General Purpose Analog Computer (GPAC) [77, 35], which is the mathematical model for analog computers (differential analyzers) used before the advent of digital computers [20] (see also the chapter "A Survey on Analog Models of Computation" in this handbook). The following result is from [13] and shows that the solution of a polynomial IVP can be computed in polynomial time with respect to a bound of $Y(t) = \max_{0\leq u\leq t} \|y(u)\|$.

Theorem 3.3.11. *There exists an algorithm $\mathscr{A}$ which on any vector of polynomials p with polynomial-time computable coefficients, $\mu \in \mathbb{N}$, $t \in \mathbb{Q}$ with $t \geq 0$, and $Y \in \mathbb{Q}$ such that $Y \geqslant \max_{0\leq u\leq t} \|y(u)\|$, satisfies*

$$\|\mathscr{A}(p,\mu,t,Y) - y(t)\|_\infty \leqslant 2^{-\mu}$$

where y is the solution of (3.5). Furthermore $\mathscr{A}(p,\mu,t,Y)$ is computed in time polynomial in the value of μ, t, and Y.

This result can be extended to IVPs where f is not necessarily a polynomial, assuming polynomial-time computability of the higher derivatives of f and an appropriate (polynomial) bound on the growth of those derivatives (see [13, Section 7]).

We note that Theorem 3.3.11 can be extended to the case where the initial condition is of the form $y(t_0) = y_0$, where $t_0 \in \mathbb{Q}$ and y_0 is polynomial-time computable (and it is actually stated in this way in the original paper [13]). This result is proved by using a variable-order method in which the solution of (3.5) is computed over a succession of subintervals $[a_i, a_{i+1}]$, with $i = 0, 1, 2, \ldots$ and $a_0 = 0$, whose union gives the maximal interval of definition of the solution y of (3.5). On each subinterval a Taylor approximation of the solution is computed, but using a variable-order method: instead of using an approximation of fixed order for each subinterval $[a_i, a_{i+1}]$, the order of the approximation is allowed to change on each interval. Note that the usual methods for numerical integrations (including basic Euler's method, etc.) fall in the general theory of n-order methods for some n. It was already suggested in [79, Section 3] that fixed-order methods might only run in exponential time when solving (3.5) for certain polynomial-time computable functions, but that variable-order methods might solve the same problem in polynomial time. Variable-order methods have also been used in certain contexts to solve IVPs, but usually without complexity results or with complexity results valid only for compact sets, see, e.g., [27, 45, 8, 1].

The previous result has the drawback that we need to know a bound $Y \geqslant \max_{0\leq u\leq t} \|y(u)\|$ (which is used as input) to be able to compute $y(u)$ for $u \in [0,t]$.

This is improved in [64], where T and Y are replaced by a single parameter – the length of the solution curve y, with the added benefit that this parameter is not needed as an input to the algorithm. We recall that the length of the curve defined by the graph of a function f between $x = a$ and $x = b$ is

$$\text{length} = \int_a^b \sqrt{1 + (f'(x))^2}\mathrm{d}x$$

In the case of the solution of (3.5) where $f = p$ is a vector of polynomials (i.e., each component of the vector p is a polynomial), we note that the derivative of the solution y is given by $p(y)$. If the degree of p is k (in the case where p has more than one component, the degree of p is the maximum of the degrees of its components), the length of the solution has a value which has an order of magnitude similar to the following quantity

$$\text{Len}(t) = \int_0^t \Sigma p \max(1, \|y(u)\|)^k \mathrm{d}u$$

where Σp denotes the sum of the absolute values of all the coefficients of p (or the maximum of such sums taken over each component of p, if p has more than one component). The following theorem is from [64].

Theorem 3.3.12. *There exists an algorithm $\mathscr{B}$ such that if $t_0, t \in \mathbb{R}$ with $t_0 \leq t$, $\varepsilon > 0$, and y satisfies (3.2) over $[t_0, t]$ and if*

$$x = \mathscr{B}(t_0, y_0, p, t, \varepsilon)$$

then $\|x - y(t)\| \leqslant \varepsilon$ and the algorithm finishes in time

$$O((\text{poly}\,(k, \text{Len}(t_0, t), \log \|y_0\|, \log \Sigma p, -\log \varepsilon))^n)$$

where $\text{Len}(t_0, t) = \int_{t_0}^t \Sigma p \max(1, \|y(u)\|)^k \mathrm{d}u$, k is the degree of p, n is the number of components of p, and $\text{poly}(a_1, \ldots, a_j)$ means polynomial in $a_1, \ldots, a_j$.

More recently [85], [52] a similar result was established for IVPs with the form (3.2), where f is analytic, using techniques similar to those of [13] (i.e., using power series to compute the solution of the IVP). The authors also characterize the complexity of solving (3.2) using parametrized complexity. In particular, if f is polynomial-time computable in several input quantities (such as a bound for a complex extension of f; the inverse of the distance to a singularity; a bound on the norm of the point where f is evaluated; the precision up to which f must be calculated), then the solution of the IVP (3.2) is also polynomial-time computable on those inputs. See [85, Theorem 4.4.1] for more details.

In [85], [54] the above result is applied to some classes of volume-preserving systems such as Hamiltonian systems. In particular, for some classes of such systems, it can be shown that their average time complexity is polynomial-time computable, provided that hard instances are relatively rare. The authors also apply this result to show that the planar circular restricted three-body problem is polynomial-time computable on average under certain assumptions.

3.4 Computability of Qualitative Behaviors of Ordinary Differential Equations

Qualitative analysis of asymptotic (long-term) behaviors is a key tool for exploring ordinary differential equations because most ODEs do not have representation formulas for their solutions and numerically computed solutions are usually limited to the short term only. In some circumstances, in particular for hyperbolic systems, qualitative analysis may also provide algorithmic descriptions, to various degrees, of certain asymptotic behaviors. There are only a few results of this nature though, which are mainly about computation of trajectory behaviors near a hyperbolic equilibrium point.

We note that a statistical approach can also be used to obtain information about the asymptotic behavior of a dynamical system. However, since this approach involves measure theory, which is discussed in its own dedicated chapter of this book, results of this nature are not discussed in this chapter.

The asymptotic behavior of a differential equation is captured by its limit sets, which are the states the solutions reach after an infinite amount of time has passed. A limit set can be a point, a finite set of points, a curve, a manifold, or even a complicated set with a fractal structure known as a strange attractor. Limit sets are well understood in dimensions one and two: in dimension one and for continuous f in (3.1), the only possible limit sets are equilibrium points; in dimension two and for C^1 smooth f, a closed and bounded limit set other than a periodic orbit or an equilibrium point consists of equilibria and solutions connecting them according to the Poincaré-Bendixson Theorem, a celebrated result in the field of dynamical systems. Structures of limit sets in dimensions greater than two become much richer and more complex; for instance, strange attractors abound in $\mathbb{R}^3$.

An equilibrium point is the simplest type of limit set.

Definition 3.4.1. 1. A point $\tilde{y} \in \mathbb{R}^n$ is called an equilibrium point of (3.1) if $f(\tilde{y}) = 0$.

2. An equilibrium point $\tilde{y}$ is called hyperbolic if none of the eigenvalues of $Df(\tilde{y})$ has zero real part, where $Df(\tilde{y})$ is the Jacobian of f at $\tilde{y}$. If the real part of every eigenvalue of $Df(\tilde{y})$ is negative (positive, respectively), then $\tilde{y}$ is called a sink (source, respectively).

The structure of solutions, also known as trajectories or orbits, near a hyperbolic equilibrium is characterized by the Stable Manifold Theorem, which is one of the most important results concerning the local qualitative theory of differential equations. From the perspective of computation, hyperbolicity ensures that Df does not have a computational singularity near $\tilde{y}$, an indication of the possible existence of an algorithmic characterization for the orbit structure near $\tilde{y}$. Note that Df presents the directions of the trajectories. We recall that the condition $x = 0$ for computable real numbers is considered a computational singularity because it cannot be decided effectively. The following theorem is an effective version of the Stable Manifold Theorem, which demonstrates that the trajectory structure near a hyperbolic equilibrium is indeed computable.

Theorem 3.4.2. *[Effective Stable Manifold Theorem [37]] Assume that $\tilde{y} \in \mathbb{R}^n$ is a hyperbolic equilibrium point of (3.1). Let $\phi(t, y_0)$ (or $\phi_t(y_0)$) denote the solution of (3.1) with the initial value y_0 at $t = 0$. Then there is a (Turing) algorithm that computes, on input $(f, Df, \tilde{y})$, a k-dimensional manifold $S \subset \mathbb{R}^n$, an $(n-k)$-dimensional manifold $U \subset \mathbb{R}^n$ with $0 \leq k \leq n$, both S and U containing $\tilde{y}$, and three positive rational numbers γ, ε, and δ such that*

(i) For all $y_0 \in S$, $\lim_{t\to+\infty} \phi(t, y_0) = \tilde{y}$ and for all $y_0 \in U$, $\lim_{t\to-\infty} \phi(t, y_0) = \tilde{y}$;
(ii) $|\phi(t, y_0) - \tilde{y}| \leq \gamma 2^{-\varepsilon t}$ for all $t \geq 0$ whenever $y_0 \in S$ and $y_0 \in B(\tilde{y}, \delta)$; $|\phi(t, y_0) - \tilde{y}| \leq \gamma 2^{\varepsilon t}$ for all $t \leq 0$ whenever $y_0 \in U$ and $y_0 \in B(\tilde{y}, \delta)$.

Moreover, if $k < n$ ($k > 0$), then a rational number η and a ball D centered at $\tilde{y}$ can be computed from f such that for any solution $\phi(t, y_0)$ to the equation (3.1) with $y_0 \in D \setminus S$, $\{\phi(t, y_0) : t \geq 0\} \not\subset B(\tilde{y}, \eta)$ ($y_0 \in D \setminus U$, $\{\phi(t, y_0) : t \leq 0\} \not\subset B(\tilde{y}, \eta)$, respectively) no matter how close the initial value y_0 is to $\tilde{y}$.

The classical proof of the Stable Manifold Theorem relies on the Jordan canonical form of $A = Df(\tilde{y})$. To reduce A to its Jordan form, one needs to find a basis of generalized eigenvectors. Since the process of finding eigenvectors from corresponding eigenvalues is not continuous in general, it is a non-computable process. Thus if one wishes to construct an algorithm that computes some S and U of (3.1) at y_0, a different method is needed. In [37] an analytic, rather than algebraic, approach to the eigenvalue problem is used to allow the computation of S and U without calling for eigenvectors. The analytic approach is based on function-theoretical treatment of resolvents (see, e.g., [84, 46, 47, 70]).

The local stable and unstable manifolds can be extended to the global stable and unstable manifolds by taking

$$W^s_f(y_0) = \bigcup_{j=0}^{\infty} \phi_{-j}(S), \quad W^u_f(y_0) = \bigcup_{j=0}^{\infty} \phi_j(U) \tag{3.7}$$

respectively. Is the global stable or unstable manifold computable for a computable hyperbolic equilibrium point? Since the global stable manifold of a sink s coincides with the basin of attraction of s, a less ambitious question is whether the basin of attraction of a computable sink is computable. The answer depends on the function f in (3.1). It is known that for hyperbolic rational functions, there are (polynomial-time) algorithms for computing basins of attraction and their complements (Julia sets) with arbitrary precision [9]; in other words, basins of attraction and Julia sets of hyperbolic rational functions are (polynomial-time) computable. On the other hand, for more general but still well-behaved functions f, the answer can be negative, even for computable and analytic functions (see [59, 100] for examples of non-computable basins of attraction occurring in C^k- and C^∞-systems). The following theorem (taken from [33]) shows that the basin of attraction of a computable sink in a computable and analytic system can be non-computable.

Theorem 3.4.3. *There exists an ODE (3.1) defined by an analytic and computable function f which admits a computable hyperbolic sink s such that the global stable manifold (basin of attraction) of s is not computable.*

Although the basin of attraction constructed in the theorem above is not computable, it is an r.e. open subset of $\mathbb{R}^n$ (see [100]); in other words, it can be filled up by a computable sequence of open balls. In general, it can be shown (see [37, Section 5]) that the degree of unsolvability of computing a global stable manifold is essentially Σ_2^0.

The proof of Theorem 3.4.3 is fairly complicated. It starts by encoding a well-known non-decidable problem into the basin of a computable hyperbolic sink in a discrete-time system; then this discrete-time system is embedded into a continuous-time system. The standard suspension method (see Smale [78], Arnold and Avez [6]) for embedding a discrete-time system into a continuous-time system is not sufficient for the construction; instead, a new method is developed for the embedding.

Another problem that is close in spirit to the Stable Manifold Theorem concerns linearization of the flow of a nonlinear ODE (3.1) near a hyperbolic equilibrium point. Poincaré originated the study and, later, a number of researchers contributed to progress on related problems, which resulted in the Hartman-Grobman Theorem (see, e.g., [63, Section 2.8]); the theorem asserts that near a hyperbolic equilibrium point $\tilde{y}$ of a nonlinear ODE (3.1), there is a homeomorphism H such that $H \circ \phi = L \circ H$, where ϕ is the solution to the ODE (3.1) and L is the solution to its linearization $\dot{y} = Df(\tilde{y})\,y$. In other words, near $\tilde{y}$, the ODE (3.1) has the same qualitative structure as its conjugated linear system. This theorem remains important, since it shows the structural stability of hyperbolic equilibria in sufficiently smooth dynamical systems. The computability study of this linearization is motivated by one of the seven open problems in the addendum to the book *Computability in Analysis and Physics* by Pour-El and Richards [67], who asked, "What is the connection between the computability of the original nonlinear operator and the linear operator which results from it?" It is proved in [38] that the homeomorphism H that performs the linearization is computable. The precise statement is given below.

For simplicity only the linearization near the origin is considered. Let $\mathscr{A}_H$ denote the set of all hyperbolic $n \times n$ matrices, where a matrix A is hyperbolic if all of its eigenvalues have nonzero real part. The operator norm is used for $A \in \mathscr{A}_H$; i.e., $\|A\| = \sup_{|x|\neq 0} |Ax|/|x|$. For Banach spaces X and Y, let $C^k(X;Y)$ denote the set of all continuously k-times differentiable functions defined on open subsets of X with ranges in Y, and $\mathscr{L}(X;Y)$ the set of all bounded linear maps from X to Y. Let $\mathscr{O}$ denote the set of all open subsets of $\mathbb{R}^n$ containing the origin of $\mathbb{R}^n$, $\mathscr{I}$ the set of all open intervals of $\mathbb{R}$ containing zero, and $\mathscr{F}$ the set of all functions $f \in C^1(\mathbb{R}^n;\mathbb{R}^n)$ such that the domain of f is in $\mathscr{O}$, $f(0) = 0$, and $Df(0) \in \mathscr{A}_H$. In other words, for any $f \in \mathscr{F}$, 0 is a hyperbolic equilibrium point of f.

Theorem 3.4.4. *[Effective Hartman-Grobman Theorem [38]] There is a computable map $\Theta : \mathscr{F} \to \mathscr{O} \times \mathscr{O} \times C(\mathbb{R}^n;\mathbb{R}) \times C(\mathbb{R}^n;\mathbb{R}^n)$ such that for any $f \in \mathscr{F}$, $f \mapsto (U,V,\mu,H)$, where*

(a) $H : U \to V$ is a homeomorphism;

(b) the unique solution $y(t,y_0) = y(y_0)(t)$ *to the initial value problem* $\dot{y} = f(y)$ *and* $y(0) = y_0$ *is defined on* $(-\mu(y_0), \mu(y_0))$ *for all* $y_0 \in U$*; moreover,* $y(t,y_0) \in U$ *for all* $y_0 \in U$ *and* $-\mu(y_0) < t < \mu(y_0)$*;*
(c) $H(y(t,y_0)) = e^{Df(0)t}H(y_0)$ *for all* $y_0 \in U$ *and* $-\mu(y_0) < t < \mu(y_0)$*.*

Recall that for any $y_0 \in \mathbb{R}^n$, $e^{Df(0)t}y_0$ is the solution to the linear problem $\dot{y} = Df(0)y$, $y(0) = y_0$. So the theorem shows that the homeomorphism H, computable from f, maps trajectories near 0, a hyperbolic equilibrium point, of the nonlinear problem $\dot{y} = f(y)$ onto trajectories near the origin of the linear problem $\dot{y} = Df(0)y$. In other words, H is a conjugacy between the linear and nonlinear trajectories near the origin. Since the classical proofs of the Hartman-Grobman linearization theorem are not constructive, the effective version of the theorem cannot be obtained from a classical proof.

The results discussed so far are centered at hyperbolic equilibrium points. Evidently, hyperbolicity and locality (in the spatial sense) are two key conditions for obtaining algorithmic descriptions about asymptotic behaviors of trajectories of the ODE (3.1) near an equilibrium; on the other hand, the role played by the structure f, near an equilibrium, of the ODE (3.1) turns out not to be as crucial, since the computations for a local stable manifold, a local unstable manifold, and the conjugacy H at a hyperbolic equilibrium are all uniform in f.

Equilibrium points are the simplest limit sets, and there are many limit sets other than equilibrium points. What about computability of other limit sets, in particular, those with complicated fractal structures such as strange attractors? The question is largely open. There is however one type of strange attractors – geometric Lorenz attractors – that has been proved to be computable recently. The Lorenz attractor is perhaps the most famous strange attractor. It was introduced in 1963 by E.N. Lorenz as one of the first examples of *strange attractors* (see, e.g., [41, 44] for a treatment of the Lorenz attractor and [4] for a more in-depth analysis of Lorenz-like attractors). However, Lorenz's research was mainly based on (non-rigorous) numerical simulations and, until recently, the proof of the existence of the Lorenz attractor remained elusive. To address that problem, Afraimovich, Bykov, and Shil'nikov [3], and Guckenheimer and Williams [42] originated the study of flows satisfying a certain list of geometric properties based on the behavior observed in the numerical simulations of the Lorenz equation. In particular, they proved that any such flow must contain a strange attractor. These examples came to be known as geometric Lorenz models, and the strange attractor contained in a geometric Lorenz flow is called the geometric Lorenz attractor. In 2002 Tucker [86] used a combination of normal form theory and rigorous numerics to provide a formal proof of the existence of the Lorenz attractor by showing that geometric Lorenz models do indeed correspond to the Lorenz system for certain parameters. Since a geometric Lorenz model supports a strange attractor, so does the Lorenz system. Recently, the following result concerning computability of geometric Lorenz attractors was proved in [40].

Theorem 3.4.5. *For any geometric Lorenz flow, if the data defining the flow are computable, then its attractor is a computable subset of* $\mathbb{R}^3$. *Moreover, the physical measure supported on this attractor is a computable probability measure.*

We should also note that some related problems can be found in verification theory. For example, in [24] the reachability problem is analyzed from a computability perspective. The reachability problem consists of finding the set of all points which can be reached in finite time from trajectories starting in an initial set. It is shown in [24] that the reachable set is lower-computable for ODEs (or, more generally, differential inclusions). However computability only holds under certain conditions.

3.5 Computability of Partial Differential Equations

The area of partial differential equations is vast and no general theory is known concerning the solvability of all PDEs. In fact, most PDEs do not even have classical solutions (a solution of a PDE of order k is classical if it is at least k times continuously differentiable in the uniform norm). Instead, they may only be solvable in the sense of weak or generalized solutions defined in some function spaces equipped with an energy-type norm such as Sobolev spaces or in the space of generalized functions. Computability theories of these spaces are established in [98, 101]. Many modern approaches to the subject – energy methods within Sobolev spaces, the calculus of variations, conservation laws, semigroups, etc. – are also useful tools in the recursion-theoretic study of the subject; in particular, energy methods augmented by a fixed-point argument.

An equation is exactly solvable if there is an explicit formula for its (classic or weak) solution. Only a few solvable partial differential equations are exactly solvable. Pour-El and Richards have studied three important second-order linear exactly solvable PDEs from the viewpoint of computability [67]: the heat equation, Laplace's equation, and the wave equation. They showed, by exploiting the solution formulas, that the heat equation and Laplace's equation admit directly computable classical solutions, at least in the case where good domains such as rectangular regions with computable corners are considered. For the wave equation, they demonstrated that a computable initial datum may generate a non-computable wave propagation measured in the uniform norm (also see [66, 68]. A discussion of those results can also be found in [31]); but in a more physically relevant setting, the wave equation does admit computable solutions in the energy norm (see, for example, [92, 88, 91]). This study indicates that computability of the solution operator of a PDE, if the PDE is uniquely solvable, depends not only on the structure of the equation but also on the physical measurements used for the initial/boundary function(s) and for the solution. Another notable fact revealed by the study is that the uniqueness of solution of a computable initial value problem for a PDE doesn't guarantee the solution being computable – a significant distinction between the computability theories of ODEs and PDEs. A related result can be found in [22], where the difficulty of solving (ordinary and partial) differential equations is studied using in-

dex sets. Several cases of differential equations are considered (e.g., equations with locally computable solutions) and are shown to correspond to certain levels in the arithmetical hierarchy, which depend on the particular problem being considered. This type of results provides more precise characterizations of non-computability results for differential equations presented in [65] and [66].

Since no general theory is known concerning the solvability of all partial differential equations, research in the field focuses on various particular PDEs which are important for applications within and outside of mathematics; for example, the KdV equation, the Schrödinger equation, and the Navier-Stokes equation. These important equations also lie at the center of the computability study of nonlinear PDEs. Indeed, as one of the seven open problems included in the Addendum in their 1989 book Computability in Analysis and Physics, Pour-El and Richards suggested: "A third area of study is the recursion-theoretic study of popular nonlinear problems of classical importance. Examples are the Navier-Stokes equation, the KdV equation, and the complex of problems associated with Feigenbaum's constant." The two theorems below give partial answers to the open question.

The first result concerns the KdV equation, which is a mathematical model of waves on shallow water surfaces. The initial value problem of the KdV equation posed on the whole real line $\mathbb{R}$:

$$\begin{cases} u_t + uu_x + u_{xxx} = 0,\ t,x \in \mathbb{R} \\ u(x,0) = \varphi(x) \end{cases}$$

defines a nonlinear map (the solution operator) $K_{\mathbb{R}}$ from the Sobolev space $H^s(\mathbb{R})$ to the space $C(\mathbb{R};H^s(\mathbb{R}))$ for real numbers $s \geq 0$, where u represents the amplitude of the wave. It is shown in [94, 93] that for any integer $s \geq 3$, the map $K_{\mathbb{R}} : H^s(\mathbb{R}) \to C(\mathbb{R};H^s(\mathbb{R}))$ is Turing computable, which means that the solution $K_{\mathbb{R}}(\varphi)$ can be computed with arbitrary precision on Turing machines when sufficiently good approximations to the initial function φ are available (in [30] a preliminary computability result was presented for the periodic case). Note that the solution operator $K_{\mathbb{R}}$ is defined for real numbers $s \geq 0$ while the computability of $K_{\mathbb{R}}$ is established only for integers $s \geq 3$ (the proof can be extended to real numbers $s \geq 3$ with some modifications). The stronger smoothness requirement is used for ensuring that several derivatives in the construction of approximations remain computable. It would be very interesting to know whether $K_{\mathbb{R}}$ remains computable for real numbers $0 \leq s < 3$.

We also note that computability results exist for similar PDEs. For example, in [99] the initial and boundary value problem (IBVP)

$$\begin{cases} u_t + u_x + uu_x + u_{xxx} = 0, \qquad t,x \geq 0 \\ u(x,0) = \varphi(x), u(0,t) = h(t) \end{cases}$$

is considered; it is shown that the operator which maps the initial and boundary data to the solution of the IBVP is computable.

The (incompressible) Navier-Stokes Equation

$$\partial_t \mathbf{u} - \triangle \mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla P = \mathbf{f}, \quad \nabla \mathbf{u} = 0, \quad \mathbf{u}(0) = \mathbf{a}, \quad \mathbf{u}\big|_{\partial\Omega} \equiv \mathbf{0} \tag{3.8}$$

describes the motion of a viscous incompressible fluid filling a rigid box $\overline{\Omega}$. The vector field $\mathbf{u} = \mathbf{u}(\mathbf{x},t) = \left(u^{(1)}, u^{(2)}, \ldots, u^{(d)}\right)$ represents the velocity of the fluid, $P = P(\mathbf{x},t)$ is the scalar pressure, and $\mathbf{f}$ is a given external force. The question of global existence and smoothness of solutions of Equation (3.8), even in the homogeneous case $\mathbf{f} \equiv \mathbf{0}$, is one of the seven Millennium Prize Problems posted by the Clay Mathematics Institute at the beginning of the twenty-first century. Local existence has been established, though, in various L^p settings [32], and uniqueness of weak solutions in dimension 2, but not in dimension 3 [80, 14]. Nevertheless, numerical solution methods have been devised in abundance, often based on pointwise (or even uniform, rather than L^p) approximation and struggling with unphysical artifacts [62]. Thus, it becomes useful to know whether it is possible to compute a local solution with arbitrary precision in a rigorous mathematical/physical setting. A recent result provides a positive answer: it is shown in [83] that the solution operator is indeed Turing computable. More precisely, it is shown that there exist a computable map T,

$$T : L^{\sigma}_{2,0}(\Omega) \times C([0,\infty), L^{\sigma}_{2,0}(\Omega)) \to (0,\infty), (\mathbf{a},\mathbf{f}) \mapsto T(\mathbf{a},\mathbf{f})$$

and a computable map $\mathscr{S}$,

$$\mathscr{S} : L^{\sigma}_{2,0}(\Omega) \times C([0,\infty), L^{\sigma}_{2,0}(\Omega)) \to C([0,\infty), L^{\sigma}_{2,0}(\Omega) \times L_2(\Omega))$$

such that for every $\mathbf{a} \in L^{\sigma}_{2,0}(\Omega)$ and $\mathbf{f} \in C([0,\infty), L^{\sigma}_{2,0}(\Omega))$, the restriction

$$\mathscr{S}(\mathbf{a},\mathbf{f})\,|_{[0,T(\mathbf{a},\mathbf{f})]} = (\mathbf{u},P)$$

constitutes a (strong local) solution to Equation (3.8), where $\Omega = (-1,1)^2$ and $L^{\sigma}_{2,0}(\Omega)$ is the set of all square-integrable, divergence-free and boundary-free functions defined on Ω.

The proofs of both theorems make use of energy methods augmented by a fixed-point argument by converting the PDE into an integral equation, assembling an iterate scheme, and then proving that the sequence generated by the iteration is computable as well as effectively convergent and the unique limit is the computable solution sought after. There are several typical difficulties in such constructions. Firstly, there is often a need for some custom-designed representations for the spaces of input and output functions, which are different from known canonical ones. For example, the weak solution of the Navier-Stokes equation (3.8) defined on a two-dimensional good region is a locally square-integrable, solenoidal (i.e., divergence-free), and boundary-free vector field. The canonical representation for an L^p space using polynomials with rational coefficients is not rich enough to encode a boundary-free solenoid. Yet a strengthened representation for the boundary-free solenoids must be sufficiently general and robust so that the information carried by the representation won't be destroyed by elementary analytical operations such as (distributive) differentiation and (primitive) integration. The second challenging

issue is that the iteration usually involves several operations, which may move back and forth among different spaces. Additional care for representations may become inevitable in order to keep the computations flowing in and out from one space to another. Thirdly, the iterate sequence may (effectively) converge only in a special subspace of the space where the sequence is defined, which may further complicate matters. For instance, Bourgain's space is needed in showing that the iterate sequence is contracting and therefore converges to the solution of the KdV equation.

In addition to particular equations, there are studies on computability aspects of several classes of PDEs. One such class is linear parabolic equations. An initial value problem of a linear parabolic equation can be written in the form:

$$\begin{cases} \mathrm{d}u(t)/\mathrm{d}t + \mathscr{A}u(t) = f(t), & t > 0 \\ u(0) = x, & x \in X \end{cases} \tag{3.9}$$

where $f : [0,T] \to X$ is a continuous map and $\mathscr{A}$ is a strong elliptic operator, bounded or unbounded, on a Banach space X. When $X = L^2(\mathbb{R}^n)$, the solution to the initial value problem (3.9) is given by the formula:

$$u(t) = W(t)x + \int_0^t W(t-s)f(s)\mathrm{d}s$$

where $W(t)$ is a C_0 semigroup generated by the infinitesimal generator $\mathscr{A}$. Thus if the semigroup W is computable from $\mathscr{A}$, then the solution operator $\mathscr{K} : (\mathscr{A}, f, x) \mapsto u$ is computable uniformly from the data $(\mathscr{A}, f, x)$ defining the problem. The study concerning computability of semigroups is carried out in [96, 97]. It is shown in [96] how to compute on Turing machines a uniformly continuous or strongly continuous semigroup uniformly from its infinitesimal generator and vice versa.

Another class for which a master algorithm exists for computing (local) solutions of any of its members is the class of general (nonlinear) first-order PDEs for scalar functions with several independent variables. A problem in this class can be written in the form:

$$F(Du, u, x) = 0 \quad \text{in } U \tag{3.10}$$

subject to the boundary condition

$$u = g \quad \text{on } \partial U \text{ (the boundary of } U) \tag{3.11}$$

where $x \in U$, U is an open subset of $\mathbb{R}^n$, $F : \mathbb{R}^n \times \mathbb{R} \times V \to \mathbb{R}$ and $g : \partial U \to \mathbb{R}$ are given, V is an open subset of $\mathbb{R}^n$ containing $\overline{U}$, the closure of U, and $u : \overline{U} \to \mathbb{R}$ is the unknown and Du is the gradient of u. A function u is called a (strong) solution of the boundary value problem if it solves (3.10), (3.11). A special feature of equation (3.10) is that it can, locally at least, be transformed into a system of first-order ordinary differential equations, called the characteristic equations, as follows: to compute the solution $u(x)$ at a point $x \in U$, one finds some curve $\mathbf{x}(s)$ lying within U, connecting x with a point $x^0 \in \partial U$, such that along the curve $\mathbf{x}(s)$ the boundary value problem for the PDE is reduced to an initial value problem for the characteristic equations. As discussed in Section 3.2, a system of ODEs is in prin-

ciple algorithmically uniformly solvable, sometimes quite explicitly. It is shown in [82] that the method of characteristics can be used to compute local solutions of (3.10), (3.11), at least for feasible instances and when the boundary ∂U is effectively locally C^1. In the same article, it is also shown that although the solution is locally computable the maximal region of existence of a computable local solution may not be computable. Moreover, the problem whether a boundary value problem has a global solution is not algorithmically decidable. The negative results hold even within the class of quasilinear equations defined by analytic computable functions over particularly simple domains (quasilinear equations are among the simplest first-order nonlinear partial differential equations). This fact shows that the algorithmic unsolvability is intrinsic.

In [15] uniform computability results are given for some simple one-dimensional elliptic boundary value problems. In particular, computability results are given for problems with the following format: (i) $-u'' + u' + u = f$ on $[0,1]$, with $u'(0) = 0$ and $u'(1) = 0$; (ii) $-u'' + au' + u = f$ on $[0,1]$, with $u(0) = u(1) = 0$; and (iii) $-u'' = f$ on $[0,1]$, with $u(0) = u(1) = 0$. The computability of the solutions of these problems follows from computable versions of the Fréchet-Riesz Theorem and the Lax-Milgram Theorem, which are also proved in [15].

In [53] the computational complexity of the Dirichlet problem for Poisson's equation

$$\Delta u = f \text{ in } \Omega,\ u|_{\partial\Omega} = g$$

is analyzed, where $\Omega \subseteq \mathbb{R}^d$ is a domain, $f : \Omega \to \mathbb{R}$ and $g : \partial\Omega \to \mathbb{R}$ are given functions, and Δ is the Laplace operator. It is shown that, for fixed d and for $\Omega = B(0,1) = \{x \in \mathbb{R}^d : \|x\| \leq 1\}$, solving the Dirichlet problem for Poisson's equation is "complete" for the complexity class $\#P$ in the sense that $\mathscr{F}\mathscr{P} = \#P$ if and only if the solution of the Dirichlet problem for Poisson's equation on $B(0,1)$ is computable in polynomial time for any choice of polynomial-time computable functions f and g.

In [95] (see also [89, 90]) the computability of the initial value problems for the linear Schrödinger equation $u_t = i\Delta u + \phi$ and the nonlinear Schrödinger equation $iu_t = -\Delta u + mu + |u|^2 u$ is studied and it is shown that the solution operator is computable if the initial data are Sobolev functions but noncomputable in the linear case if the initial data are L^p-functions and $p \neq 2$.

In [72] the authors study symmetric hyperbolic systems of PDEs of the form

$$\begin{cases} A\frac{\partial u}{\partial t} + \sum_{i=1}^{m} B_i \frac{\partial u}{\partial x_i} = 0 \\ u(x,0) = \varphi(x) \end{cases} \tag{3.12}$$

where $A, B_1, \ldots, B_m$ are computable constant symmetric $n \times n$ matrices, $t \geq 0$, and $x \in [0,1]^m$. There it is shown that if the matrices are fixed and if the first- and second-order derivatives of the initial function φ are bounded, the operator which maps the initial condition φ to the solution of the PDE is computable (see also [73] for a correction and [74, Remark 5.5.3] for the statement given here, which removes some unneeded assumptions used in the original formulation of [72]). Later it was shown in [74, Theorem 5.3] that the operator which maps an initial condition and the

matrices $A, B_1, \ldots, B_m$ to the solution of the PDE is uniformly computable, provided the matrices satisfy some algebraic conditions.

More recently, the computational complexity of solving PDEs with the format (3.12) was studied in [76, 75]. There the (uniform) complexity of this problem is shown to be in EXPTIME in general and in PTIME under certain conditions.

The Dirichlet Problem was also analyzed in [16, 17, 18] using Bishop's framework of constructive mathematics. It is shown that weak solutions of the Dirichlet Problem exist if certain conditions hold.

3.6 Some Open Problems

In this chapter, we have presented several results concerning the computability and computational complexity of differential equations. However, the field of differential equations is vast, and many more problems in the field await exploration from the perspectives of computable analysis. Here we merely give two open problems.

1. The first problem is to give general principles from which the computability or noncomputability of limit sets of ODEs follows as corollaries. For example, it has been shown that a basin of attraction of an ODE defined by a hyperbolic rational function f is computable, uniformly in f ([9]); however, on the other hand, if f is analytic, computable but non-hyperbolic, a basin of attraction is no longer necessarily computable ([33]). It would be interesting to know whether a computable, analytic and hyperbolic f ensures the computability of a basin of attraction.
2. The second problem is to characterize the computational complexity of well-known computable PDEs such as the KdV equation, the Schrödinger equation, and the Navier-Stokes equation.

Acknowledgements D. Graça was partially funded by FCT/MCTES through national funds and co-funded EU funds under the project UIDB/50008/2020. This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 731143.

References

1. Abad, A., Barrio, R., Blesa, F., Rodríguez, M.: Algorithm 924: Tides, a Taylor series integrator for differential equations. ACM Trans. Math. Softw. **39**(1), 5:1–5:28 (2012)
2. Aberth, O.: The failure in computable analysis of a classical existence theorem for differential equations. Proceedings of the American Mathematical Society **30**, 151–156 (1971)
3. Afraimovich, V.S., Bykov, V.V., Shil'nikov, L.P.: On the appearence and structure of the Lorenz attractor. Dokl. Acad. Sci. USSR **234**, 336–339 (1977)
4. Araújo, V., Pacifico, M.: Three-Dimensional Flows. Springer (2010)
5. Arnold, V.I.: Ordinary Differential Equations. MIT Press (1978)

6. Arnold, V.I., Avez, A.: Ergodic Problems of Classical Mechanics. W.A. Benjamin (1968)
7. Atkinson, K.E.: An Introduction to Numerical Analysis, 2nd edn. John Wiley & Sons (1989)
8. Barrio, R., Rodríguez, M., Abad, A., Blesa, F.: Breaking the limits: the Taylor series method. Appl. Math. Comput. **217**(20), 7940–7954 (2011)
9. Beardon, A.F.: Iteration of Rational Functions. Springer (1991)
10. Birkhoff, G. (ed.): A Source Book in Classical Analysis, chap. Osgood's Existence Theorem, pp. 251–258. Harvard University Press (1973)
11. Birkhoff, G., Rota, G.C.: Ordinary Differential Equations, 4th edn. John Wiley & Sons (1989)
12. Bournez, O., Graça, D.S., Pouly, A.: Solving analytic differential equations in polynomial time over unbounded domains. In: F. Murlak, P. Sankowski (eds.) Mathematical Foundations of Computer Science 2011, *Lecture Notes in Computer Science*, vol. 6907, pp. 170–181. Springer, Heidelberg (2011)
13. Bournez, O., Graça, D.S., Pouly, A.: On the complexity of solving initial value problems. In: J. van der Hoeven, M. van Hoeij (eds.) ISSAC 2012—Proceedings of the 37th International Symposium on Symbolic and Algebraic Computation, pp. 115–121. ACM, New York (2012)
14. Boyer, F., Fabrie, P.: Mathematical Tools for the Study of the Incompressible Navier-Stokes Equations and Related Models. Springer (2013)
15. Brattka, V., Yoshikawa, A.: Towards computability of elliptic boundary value problems in variational formulation. Journal of Complexity **22**(6), 858–880 (2006)
16. Bridges, D.S., Wang, Y.: Constructive aspects of the Dirichlet problem. Journal of Universal Computer Science **3**(11), 1148–1161 (electronic) (1997)
17. Bridges, D.S., Wang, Y.: Constructive weak solutions of the Dirichlet problem. The Journal of the London Mathematical Society. Second Series **57**(3), 655–667 (1998)
18. Bridges, D.S., Wang, Y.: Weak solutions of the Dirichlet problem and the locatedness of $H_0^1(\omega)$. New Zealand Journal of Mathematics **27**(1), 1–5 (1998)
19. Buser, P., Scarpellini, B.: Recursive analysis of singular ordinary differential equations. Annals of Pure and Applied Logic **162**, 20–35 (2010)
20. Bush, V.: The differential analyzer. A new machine for solving differential equations. J. Franklin Inst. **212**, 447–488 (1931)
21. Carothers, D.C., Parker, G.E., Sochacki, J.S., Warne, P.G.: Some properties of solutions to polynomial systems of differential equations. Electron. J. Diff. Eqns. **2005**(40) (2005)
22. Cenzer, D., Remmel, J.B.: Index sets for computable differential equations. Mathematical Logic Quarterly **50**(4,5), 329–344 (2004)
23. Cleave, J.: The primitive recursive analysis of ordinary differential equations and the complexity of their solutions. Journal of Computer and Systems Sciences **3**, 447–455 (1969)
24. Collins, P.: Continuity and computability on reachable sets. Theoretical Computer Science **341**, 162–195 (2005)
25. Collins, P., Graça, D.: Effective computability of solutions of ordinary differential equations the thousand monkeys approach. In: V. Brattka, R. Dillhage, T. Grubba, A. Klutsch (eds.) CCA 2008, Fifth International Conference on Computability and Complexity in Analysis, *Electronic Notes in Theoretical Computer Science*, vol. 221, pp. 103–114. Elsevier (2008), CCA 2008, Fifth International Conference, Hagen, Germany, August 21–24, 2008
26. Collins, P., Graça, D.S.: Effective computability of solutions of differential inclusions: the ten thousand monkeys approach. Journal of Universal Computer Science **15**(6), 1162–1185 (2009)
27. Corliss, G., Chang, Y.F.: Solving ordinary differential equations using Taylor series. ACM Trans. Math. Softw. **8**(2), 114–144 (1982)
28. Edalat, A., Pattinson, D.: A domain-theoretic account of Picard's theorem. LMS Journal of Computation and Mathematics **10**, 83–118 (2007)
29. Farjudian, A.: Polynomial-time solution of initial value problems using polynomial enclosures. In: L. Ong, R. de Queiroz (eds.) Logic, Language, Information and Computation, *Lecture Notes in Computer Science*, vol. 7456, pp. 232–245. Springer, Heidelberg (2012)
30. Gay, W., Zhang, B.Y., Zhong, N.: Computability of solutions of the Korteweg-de Vries equation. Mathematical Logic Quarterly **47**(1), 93–110 (2001)

31. Gherardi, G.: Computability and incomputability of differential equations. In: G. Corsi, R. Lupacchini (eds.) Deduction, Computation, Experiment, pp. 223–242. Springer Verlag, Milano (2008)
32. Giga, Y., Miyakawa, T.: Solutions in L_r of the Navier-Stokes initial value problem. Archive for Rational Mechanics and Analysis **89**, 267–281 (1985)
33. Graça, D., Zhong, N.: An analytic system with a computable hyperbolic sink whose basin of attraction is non-computable. Theory of Computing Systems **57**(2), 478–520 (2015)
34. Graça, D.S., Campagnolo, M.L., Buescu, J.: Computability with polynomial differential equations. Advances in Applied Mathematics **40**(3), 330–349 (2008)
35. Graça, D.S., Costa, J.F.: Analog computers and recursive functions over the reals. Journal of Complexity **19**(5), 644–664 (2003)
36. Graça, D.S., Zhong, N., Buescu, J.: Computability, noncomputability and undecidability of maximal intervals of IVPs. Transactions of the American Mathematical Society **361**(6), 2913–2927 (2009)
37. Graça, D.S., Zhong, N., Buescu, J.: Computability, noncomputability, and hyperbolic systems. Appl. Math. Comput. **219**(6), 3039–3054 (2012)
38. Graça, D.S., Zhong, N., Dumas, H.S.: The connection between computability of a nonlinear problem and its linearization: The Hartman-Grobman theorem revisited. Theoret. Comput. Sci. **457**, 101–110 (2012)
39. Graça, D.S., Buescu, J., Campagnolo, M.L.: Boundedness of the domain of definition is undecidable for polynomial ODEs. In: R. Dillhage, T. Grubba, A. Sorbi, K. Weihrauch, N. Zhong (eds.) Proceedings of the Fourth International Conference on Computability and Complexity in Analysis (CCA 2007), *Electronic Notes in Theoretical Computer Science*, vol. 202, pp. 49–57. Elsevier (2008), CCA 2007, Siena, Italy, June 16–18, 2007
40. Graça, D.S., Rojas, C., Zhong, N.: Computing geometric Lorenz attractors with arbitrary precision. Transactions of the American Mathematical Society **370**(4), 2955–2970 (2018)
41. Guckenheimer, J., Holmes, P.: Nonlinear Oscillations, Dynamical Systems, and Bifurcation of Vector Fields. Springer (1983)
42. Guckenheimer, J., Williams, R.F.: Structural stability of Lorenz attractors. Publ. Math. IHES **50**, 59–72 (1979)
43. Hartman, P.: Ordinary Differential Equations, 2nd edn. Birkhäuser (1982)
44. Hirsch, M.W., Smale, S., Devaney, R.: Differential Equations, Dynamical Systems, and an Introduction to Chaos. Academic Press (2004)
45. Jorba, À., Zou, M.: A software package for the numerical integration of ODEs by means of high-order taylor methods. Experimental Mathematics **14**(1), 99–117 (2005)
46. Kato, T.: On the convergence of the perturbation method. i. Progr. Theor. Phys. **4**(4), 514–523 (1949)
47. Kato, T.: On the convergence of the perturbation method. ii. Progr. Theor. Phys. **5**(1), 95–101 (1950)
48. Kawamura, A.: Lipschitz continuous ordinary differential equations are polynomial-space complete. In: 24th Annual IEEE Conference on Computational Complexity, pp. 149–160. IEEE Computer Soc., Los Alamitos, CA (2009)
49. Kawamura, A.: Lipschitz continuous ordinary differential equations are polynomial-space complete. Computational Complexity **19**(2), 305–332 (2010)
50. Kawamura, A., Cook, S.A.: Complexity theory for operators in analysis. ACM Transactions on Computation Theory **4**(2), 5:1–5:24 (2012)
51. Kawamura, A., Ota, H., Rösnick, C., Ziegler, M.: Computational complexity of smooth differential equations. Logical Methods in Computer Science **10**, 1:6,15 (2014)
52. Kawamura, A., Steinberg, F., Thies, H.: Parameterized complexity for uniform operators on multidimensional analytic functions and ODE solving. In: L. Moss, R. de Queiroz, M. Martinez (eds.) Logic, language, information, and computation, *Lecture Notes in Computer Science*, vol. 10944, pp. 223–236. Springer, Berlin (2018)
53. Kawamura, A., Steinberg, F., Ziegler, M.: On the computational complexity of the Dirichlet problem for Poisson's equation. Mathematical Structures in Computer Science **27**(8), 1437–1465 (2017)

54. Kawamura, A., Thies, H., Ziegler, M.: Average-case polynomial-time computability of Hamiltonian dynamics. In: 43rd International Symposium on Mathematical Foundations of Computer Science, *LIPIcs. Leibniz Int. Proc. Inform.*, vol. 117, pp. Art. No. 30, 17. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern (2018)
55. Ko, K.I.: On the computational complexity of ordinary differential equations. Inform. Contr. **58**, 157–194 (1983)
56. Ko, K.I.: Complexity Theory of Real Functions. *Progress in Theoretical Computer Science*. Birkhäuser, Boston (1991)
57. Ko, K.I., Friedman, H.: Computing power series in polynomial time. Advances in Applied Math. **9**, 40–50 (1988)
58. Miller, W.: Recursive function theory and numerical analysis. Journal of Computer and Systems Sciences **4**, 465–472 (1970)
59. Moore, C.: Generalized shifts: unpredictability and undecidability in dynamical systems. Nonlinearity **4**(2), 199–230 (1991)
60. Müller, N.T.: Uniform computational complexity of Taylor series. In: T. Ottmann (ed.) Proceedings of the 14th International Colloquium on Automata, Languages, and Programming, *Lecture Notes in Computer Science*, vol. 267, pp. 435–444. Springer, Berlin (1987)
61. Müller, N.T., Moiske, B.: Solving initial value problems in polynomial time. In: Proceedings of the 22th JAIIO - Panel'93, Part 2, pp. 283–293 (1993), Buenos Aires, 1993
62. Patel, M.K., Markatos, N.C., Cross, M.: A critical evaluation of seven discretization schemes for convection-diffusion equations. International Journal for Numerical Methods in Fluids **5**(3), 225–244 (1985)
63. Perko, L.: Differential Equations and Dynamical Systems, 3rd edn. Springer (2001)
64. Pouly, A., Graça, D.S.: Computational complexity of solving polynomial differential equations over unbounded domains. Theoretical Computer Science **626**, 67–82 (2016)
65. Pour-El, M.B., Richards, J.I.: A computable ordinary differential equation which possesses no computable solution. Annals Math. Logic **17**, 61–90 (1979)
66. Pour-El, M.B., Richards, J.I.: The wave equation with computable initial data such that its unique solution is not computable. Advances in Math. **39**, 215–239 (1981)
67. Pour-El, M.B., Richards, J.I.: Computability in Analysis and Physics. *Perspectives in Mathematical Logic*. Springer, Berlin (1989)
68. Pour-El, M.B., Zhong, N.: The wave equation with computable initial data whose unique solution is nowhere computable. Mathematical Logic Quarterly **43**(4), 499–509 (1997)
69. Rettinger, R., Weihrauch, K., Zhong, N.: Topological complexity of blowup problems. Journal of Universal Computer Science **15**(6), 1301–1316 (2009)
70. Robinson, C.: Dynamical Systems. CRC Press (1995)
71. Ruohonen, K.: An effective Cauchy-Peano existence theorem for unique solutions. Internat. J. Found. Comput. Sci. **7**(2), 151–160 (1996)
72. Selivanova, S., Selivanov, V.: Computing solutions of symmetric hyperbolic systems of PDE's. In: V. Brattka, R. Dillhage, T. Grubba, A. Klutsch (eds.) CCA 2008, Fifth International Conference on Computability and Complexity in Analysis, *Electronic Notes in Theoretical Computer Science*, vol. 221, pp. 243–255. Elsevier (2008), CCA 2008, Fifth International Conference, Hagen, Germany, August 21–24, 2008
73. Selivanova, S., Selivanov, V.: Computing the solution operators of symmetric hyperbolic systems of PDE. Journal of Universal Computer Science **15**(6), 1337–1364 (2009)
74. Selivanova, S., Selivanov, V.: Computing solution operators of boundary-value problems for some linear hyperbolic systems of PDEs. Logical Methods in Computer Science **13**(4) (2017)
75. Selivanova, S., Selivanov, V.: Bit complexity of computing solutions for symmetric hyperbolic systems of PDEs with guaranteed precision. Computability **10**, (to appear) (2021)
76. Selivanova, S.V., Selivanov, V.L.: Bit complexity of computing solutions for symmetric hyperbolic systems of PDEs (extended abstract). In: F. Manea, R. Miller, D. Nowotka (eds.) Sailing Routes in the World of Computation, *Lecture Notes in Computer Science*, vol. 10936, pp. 376–385. Springer, Cham (2018)
77. Shannon, C.E.: Mathematical theory of the differential analyzer. J. Math. Phys. MIT **20**, 337–354 (1941)

78. Smale, S.: Differentiable dynamical systems. Bull. Amer. Math. Soc. **73**, 747–817 (1967)
79. Smith, W.D.: Church's thesis meets the N-body problem. Applied Mathematics and Computation **178**(1), 154–183 (2006)
80. Sohr, H.: The Navier-Stokes Equations: An Elementary Functional Analytic Approach. Birkhäuser (2001)
81. Sun, S.M., Zhong, N.: On effective convergence of numerical solutions for differential equations. ACM Transactions on Computation Theory **6**(1), Art. 4, 25 (2014)
82. Sun, S.M., Zhong, N.: Computability aspects for 1st-order partial differential equations via characteristics. Theoretical Computer Science **583**, 27–39 (2015)
83. Sun, S.M., Zhong, N., Ziegler, M.: On computability of Navier-Stokes' equation. In: A. Beckmann, V. Mitrana, M. Soskova (eds.) Evolving Computability, *Lecture Notes in Computer Science*, vol. 9136, pp. 334–342. Springer, Cham (2015), 11th Conference on Computability in Europe, CiE 2015, Bucharest, Romania, June 29–July 3, 2015
84. Szőkefalvi-Nagy, B.: Spektraldarstellung linearer Transformationen des Hilbertschen Raumes. Springer (1942)
85. Thies, H.: Uniform computational complexity of ordinary differential equations with applications to dynamical systems and exact real arithmetic. Graduate School of Arts and Sciences, University of Tokyo, Tokyo, Japan (2018)
86. Tucker, W.: A rigorous ODE solver and Smale's 14th problem. Foundations of Computational Mathematics **2**(1), 53–117 (2002)
87. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)
88. Weihrauch, K., Zhong, N.: The wave propagator is Turing computable. In: J. Wiedermann, P. van Emde Boas, M. Nielsen (eds.) Automata, Languages and Programming, *Lecture Notes in Computer Science*, vol. 1644, pp. 697–706. Springer, Berlin (1999), 26th International Colloquium, ICALP'99, Prague, Czech Republic, July, 1999
89. Weihrauch, K., Zhong, N.: Is the linear Schrödinger propagator Turing computable? In: J. Blanck, V. Brattka, P. Hertling (eds.) Computability and Complexity in Analysis, *Lecture Notes in Computer Science*, vol. 2064, pp. 369–377. Springer, Berlin (2001), 4th International Workshop, CCA 2000, Swansea, UK, September 2000
90. Weihrauch, K., Zhong, N.: Turing computability of a nonlinear Schrödinger propagator. In: J. Wang (ed.) Computing and Combinatorics, *Lecture Notes in Computer Science*, vol. 2108, pp. 596–599. Springer, Berlin (2001), 7th Annual International Conference, COCOON 2001, Guilin, China, August 20-23, 2001
91. Weihrauch, K., Zhong, N.: Is wave propagation computable or can wave computers beat the Turing machine? Proceedings of the London Mathematical Society **85**(2), 312–332 (2002)
92. Weihrauch, K., Zhong, N.: The solution operator of the Korteweg-de Vries equation is computable. In: V. Brattka, M. Schröder, K. Weihrauch (eds.) CCA 2002 Computability and Complexity in Analysis, *Electronic Notes in Theoretical Computer Science*, vol. 66. Elsevier, Amsterdam (2002), 5th International Workshop, CCA 2002, Málaga, Spain, July 12–13, 2002
93. Weihrauch, K., Zhong, N.: Computing the solution of the Korteweg-de Vries equation with arbitrary precision on Turing machines. Theoretical Computer Science **332**(1–3), 337–366 (2005)
94. Weihrauch, K., Zhong, N.: An algorithm for computing fundamental solutions. SIAM Journal on Computing **35**(6), 1283–1294 (2006)
95. Weihrauch, K., Zhong, N.: Computing Schrödinger propagators on Type-2 Turing machines. Journal of Complexity **22**(6), 918–935 (2006)
96. Weihrauch, K., Zhong, N.: Computable analysis of the abstract Cauchy problem in a Banach space and its applications (I). In: D. Cenzer, R. Dillhage, T. Grubba, K. Weihrauch (eds.) Proceedings of the Third International Conference on Computability and Complexity in Analysis, *Electronic Notes in Theoretical Computer Science*, vol. 167, pp. 33–59. Elsevier, Amsterdam (2007), CCA 2006, Gainesville, Florida, USA, November 1–5, 2006
97. Weihrauch, K., Zhong, N.: Computable analysis of the abstract Cauchy problem in a Banach space and its applications I. Mathematical Logic Quarterly **53**(4–5), 511–531 (2007)

98. Zhong, N.: Computability structure of the Sobolev spaces and its applications. Theoretical Computer Science **219**, 487–510 (1999)
99. Zhong, N.: Computable analysis of a non-homogeneous boundary-value problem for the Korteweg-de Vries equation. In: S.B. Cooper, B. Löwe, L. Torenvliet (eds.) New Computational Paradigms, *Lecture Notes in Computer Science*, vol. 3526, pp. 552–561. Springer, Berlin (2005), First Conference on Computability in Europe, CiE 2005, Amsterdam, The Netherlands, June 2005
100. Zhong, N.: Computational unsolvability of domains of attraction of nonlinear systems. Proc. Amer. Math. Soc. **137**(8), 2773–2783 (2009)
101. Zhong, N., Weihrauch, K.: Computability theory of generalized functions. Journal of the Association for Computing Machinery **50**(4), 469–505 (2003)

Chapter 4
Computable Complex Analysis

Valentin V. Andreev and Timothy H. McNicholl

Abstract We give an introduction to the field of computable complex analysis and survey results on conformal mapping, harmonic functions, infinite products, and constants. We also suggest problems for further investigation.

4.1 Introduction

Computable analysis is the mathematical study of computation with continuous data such as real numbers. In the appendix on open problems in their seminal text on the subject, M. Pour-El and J. Richards remark:

> Let us consider this issue from the viewpoint of complex analysis...Here, the crucial theorem is the Riemann Mapping Theorem...Two obvious questions are (1) If the mapping is computable, is the region computable? (2) If the region is computable, is the mapping computable? This in turn leads to the question of what we should mean by a computable region in the plane....The resolution of this question might provide an interesting interplay between plane set topology, complex analysis, and logic....For analytic aspects, we might consider conformal mappings of multiply connected regions. [82]

Our goal here is to give a mostly self-contained survey of the developments in the decades after the publication of their monograph. We in particular want to direct attention to how the development of computable complex analysis led to precisely the interesting interplay anticipated by Pour-El and Richards between classical analysis and computability. In particular, work on effective versions of classical theorems parallels in many cases the efforts in classical mathematics to establish constructive

Valentin V. Andreev
Department of Mathematics, Lamar University, Beaumont, Texas, USA
e-mail: valentin.andreev@lamar.edu

Timothy H. McNicholl
Department of Mathematics, Iowa State University, Ames, Iowa, USA
e-mail: mcnichol@iastate.edu

V. Brattka, P. Hertling (eds.), *Handbook of Computability and Complexity in Analysis*,
Theory and Applications of Computability, https://doi.org/10.1007/978-3-030-59234-9_4

versions. In addition, progress on the effective theory leads in some cases to consideration of new problems in classical analysis and is sometimes facilitated by recent developments in complex and harmonic function theory.

We review work in the following broad areas: conformal mapping with particular attention to the Riemann Mapping Theorem (in accordance with the direction set by the above quote), harmonic functions, infinite products, and constants. Within the broad area of conformal mapping we will also survey work on simply connected domains, multiply connected domains, and boundary extensions.

The emphasis of Pour-El and Richards on the Riemann Mapping Theorem is no doubt due to the fact that this theorem and its generalization, the Uniformization Theorem, are probably the most profound theorems of complex analysis. Informally, the Riemann mapping theorem says that every simply connected domain in the complex plane with at least two boundary points is conformally equivalent to the open unit disk, and the Uniformization Theorem says that every simply connected Riemann surface is conformally equivalent to the open unit disk, the complex plane, or the Riemann sphere. These theorems have been and continue to be the cornerstone for the research of generations of mathematicians both in theoretical and applied complex analysis.

While Riemann's proof published in 1851 in his *Inauguraldissertation* was flawed, his work inspired numerous proofs of the Riemann Mapping Theorem which developed and employed different far-reaching methods, many of which were nonconstructive. Very early, mathematicians began to search for a constructive approach to the theorem, the most notable of which were the discovery of the Schwarz-Christoffel mapping by H. A. Schwarz and E. B. Christoffel in the 1870s and parts of the proof of the Riemann mapping theorem by P. Koebe in 1912. The work we are to describe can be seen as being in the spirit of these classical directions, but sharpened with the precise mathematical theory of computability.

There are many important differences between conformal mapping of multiply connected domains (i.e., domains that are not simply connected in the sense that they have one or more 'holes') and simply connected domains. One is that where there is exactly one canonical domain in the simply connected case (namely the unit disk), there are many canonical domains in the multiply connected case. These challenges led to several developments in the classical theory, and, as noted by Pour-El and Richards, these lead to an interesting topic of investigation in computable analysis. These inquiries also engender the development of the effective theory of harmonic functions in the plane and boundary extensions of conformal maps.

Infinite products also play an important role in the development of the classical theory and often lead to numerical methods that converge more quickly than the corresponding series expansions. We will focus on Blaschke products, which play a central role in the understanding of spaces of analytic functions (see, e.g., [31, 29]).

The study of constants such as Bloch's constant, whose existence alone is surprising, is one of the other important directions of the classical theory. Whereas the decimal expansions of other constants in mathematics such as π and e can be computed, usually only a few decimals of the constants of complex analysis are known.

Thus, the study of the computability of these constants can be seen as a contribution to the classical theory.

The paper is organized as follows.

- Section 4.2: Preliminaries from complex analysis.
- Section 4.3: Computability on the complex plane and extended complex plane.
- Section 4.4: Computable conformal mapping of simply connected domains.
- Section 4.5: Boundary extensions.
- Section 4.6: Harmonic functions.
- Section 4.7: Conformal mapping of multiply connected domains.
- Section 4.8: Infinite products.
- Section 4.9: Constants.
- Section 4.10: Open problems.

To make the paper as self-contained as possible, we present some necessary ideas from complex analysis in Section 4.2 and a simple treatment of computability in the complex and extended complex plane in Section 4.3.

In Section 4.4, after some historical notes on the proof of the Riemann Mapping Theorem, we discuss the obstacles to the effectivization of the Koebe construction of 1912 and the techniques employed by Cheng, Bishop, and Bridges for the constructive proof of the theorem as well as Hertling's effective proof.

Section 4.5 is devoted to the question: under what conditions can a conformal map between two domains be extended to a continuous or even homeomorphic map of the closures of the domains. If such an extension exists, it is referred to as a *boundary extension*. If the conformal map is computable, it is also natural to ask whether its boundary extension is computable. We will survey the classical material as well as computability results.

In Section 4.6, we consider computability questions related to harmonic conjugates, analytic extensions, and Dirichlet problems.

As we mentioned earlier, there are major differences between conformal mappings on simply connected domains and conformal mappings on multiply connected domains. To begin with, in the simply connected case the unit disk plays an exceptional role due to the Riemann Mapping Theorem, while in the case of multiply connected domains there is, for example, no single circular domain that plays the role of a *canonical domain* for all circular domains of the same connectivity. These differences will be discussed in depth in Section 4.7, after which we present the Koebe circular domain construction and the work of Andreev and McNicholl on the corresponding error estimates. We also discuss Schiffer's approach, which was employed by Andreev and McNicholl to show that the conformal maps of a finitely connected domain onto the canonical slit domains can be computed uniformly from the domain and its boundary.

Section 4.8 discusses the computability of the representation of an analytic function as a Blaschke product. In Section 4.9 we encapsulate recent results of R. Rettinger on the computability of Bloch's and Landau's constants, and in the last section we put forth some open problems which we hope will inspire further research.

Again, to keep the paper as self-contained as possible, we begin each of Sections 4.4 through 4.9 with a synopsis of the related classical theory before delving into the effective theory. In some cases we also discuss complexity results and work on practical computation.

4.2 Preliminaries from Complex Analysis

We summarize here the fundamental material on complex analysis that supports our later discussions and is found in the early chapters of standard sources such as [19, 36, 33].

We denote the complex plane by $\mathbb{C}$. We let $\mathbb{D}$ denote the unit disk; that is the set of all complex numbers of modulus smaller than 1. Let $\mathbb{H}$ denote the right half-plane; i.e., the set of all complex numbers with positive real part. Let $D(z;r)$ denote the open disk with center z and radius r.

A *curve* is a continuous function from the unit interval into the complex plane. We generally identify curves with their ranges. A curve $\gamma\colon [0,1] \to$ is a *Jordan curve* if $\gamma(s) = \gamma(t)$ if and only if $s,t \in \{0,1\}$. When viewing a Jordan curve as a set, such a function γ is referred to as a *parameterization*.

Suppose γ is a Jordan curve. By the Jordan Curve Theorem, $\mathbb{C} - \gamma$ has two connected components, one of which is bounded and one of which is not. The bounded component is referred to as the *interior* of γ and will be denoted $\mathrm{Int}(\gamma)$. The unbounded component is referred to as the *exterior* of γ and will be denoted $\mathrm{Ext}(\gamma)$.

A *domain* is an open and connected subset of the complex plane. A domain is a *Jordan domain* if each connected component of its complement is bounded by a Jordan curve. A domain D is *simply connected* if D includes the interior of γ whenever γ is a Jordan curve that is included in D; otherwise it is said to be *multiply connected*.

When $f :\subseteq \mathbb{C} \to \mathbb{C}$ and $z_0 \in \mathrm{dom}(f)$, we say that f is differentiable at z_0 if

$$\lim_{z \to z_0} \frac{f(z) - f(z_0)}{z - z_0}$$

exists, in which case the value of this limit is called the *derivative of f at z_0* and is denoted $f'(z_0)$. A function $f :\subseteq \mathbb{C} \to \mathbb{C}$ is *analytic* if it is differentiable and if its domain of definition is open and connected. It follows from the Cauchy-Goursat Theorem that every analytic function is infinitely differentiable. Furthermore, if f is analytic and $D(c;R) \subseteq \mathrm{dom}(f)$, then

$$f(z) = \sum_{n=0}^{\infty} \frac{f^{(n)}(c)}{n!}(z-c)^n$$

whenever $|z - c| < R$. That is, analytic functions have series expansions at each point of their domain which are valid on the largest open disk possible. Conversely,

every function defined on an open disk by a power series is analytic. Finally, if f is analytic, and if γ is a piecewise smooth Jordan curve such that $\overline{\text{Int}(\gamma)} \subseteq \text{dom}(f)$, then by the Cauchy Integral Formula

$$f^{(n)}(c) = \frac{n!}{2\pi i} \int_\gamma f(z)\, \mathrm{d}z$$

whenever z_0 belongs to the interior of γ.

The zeros of an analytic function have a number of remarkable features. One is that they are isolated. That is, if f is a non-constant analytic function and $f(z_0) = 0$, then there is a positive number r such that $f(z) \neq 0$ whenever $0 < |z - z_0| < r$. This in turn leads to the Identity Theorem: if two analytic functions on a domain D agree on a set with an accumulation point in D, then they are in fact the same function. In addition, each zero of a non-constant analytic function has an associated multiplicity; i.e., if f is a non-constant analytic function and $f(z_0) = 0$, then there is a largest number m such that $f(z) = (z - z_0)^m \phi(z)$ for some analytic function ϕ with $\phi(z_0) \neq 0$. Finally, if f is an analytic function and if γ is a piecewise smooth Jordan curve such that $\overline{\text{Int}(\gamma)} \subseteq \text{dom}(f) \cap f^{-1}[\mathbb{C} - \{0\}]$, then the number of zeros of f inside γ is calculated by the integral

$$\frac{1}{2\pi i} \int_\gamma \frac{f'(z)}{f(z)} \mathrm{d}z.$$

This is known as the Argument Principle.

An important consequence of the Argument Principle is the Open Mapping Theorem: every analytic function is an open map. That is, if U is an open subset of the domain of an analytic function f, then $f[U]$ is open.

When D is a domain, a function $u : D \to \mathbb{R}$ is *harmonic* if it is twice continuously differentiable and if it satisfies Laplace's Equation:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0.$$

A simple but useful example of a harmonic function is $u(z) = \log |z|$. In addition, the composition of a harmonic function with an analytic function yields another harmonic function.

If f is an analytic function, then its real and imaginary parts are harmonic and satisfy the *Cauchy-Riemann equations*:

$$\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y},$$
$$\frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x}.$$

Conversely, if D is a domain, and if the real and imaginary parts of a function $f : D \to \mathbb{C}$ have continuous partials and satisfy the Cauchy-Riemann equations, then f is analytic.

If $u, v : D \to \mathbb{R}$ are harmonic functions that satisfy the Cauchy-Riemann equations, then we say that v is a *conjugate* of u and write $v = \tilde{u}$. Note that conjugates are not unique but they are unique up to an additive constant. Note also that not every harmonic function has a harmonic conjugate on D; e.g., $u(x,y) = \frac{1}{2}\ln(x^2+y^2)$. However, if D is simply connected, then every harmonic function on D has a harmonic conjugate on D. It is sometimes desired to build an analytic function on a domain D by first constructing its real part and then a conjugate of its real part. Accordingly, when $u : D \to \mathbb{R}$ is harmonic, an *analytic* extension of u is a function of the form $f = u + i\tilde{u}$. We write $f = \hat{u}$ if f is an analytic extension of u. Again, a harmonic function will have more than one analytic extension, but these extensions all differ by a constant.

An analytic map is *conformal* if its derivative is never zero. It is an easy consequence of the chain rule that such maps preserve angles between tangent lines of curves. By the Schwarz Lemma, every conformal map f of the unit disk onto itself has the form

$$f(z) = \lambda \frac{z - z_0}{1 - z\overline{z_0}}$$

for some $z_0 \in \mathbb{D}$ and $\lambda \in \partial\mathbb{D}$.

4.3 Computability on the Complex Plane and Extended Complex Plane

It is assumed that the reader is familiar with the rudiments of the classical computability theory on discrete structures as expounded in [26, 90, 95, 20]. A comprehensive treatment of computability on continuous domains can be found in [101]. See also [14, 15, 37, 46, 47, 56, 57, 58, 82, 98]. Some of these approaches are very general. In particular, the Type-Two Effectivity approach is capable of defining computability on any second-countable T_0 space [101]. For simplicity, we will limit our scope to computability on the complex plane and the extended complex plane. Our approach to computability on these domains is equivalent to what would be obtained from any of the more general frameworks.

We begin with points and sequences. We say that a point $z \in \mathbb{C}$ is *computable* if there is an algorithm that given any nonnegative integer k as input computes a rational point $q \in \mathbb{C}$ such that $|z - q| < 2^{-k}$. (By a *rational point* we mean a complex number whose real and imaginary parts are rational.) We think of q as an approximation of z and 2^{-k} as an upper bound on the error in this approximation. So, less formally, z is computable if we can compute arbitrarily good approximations of z. A sequence of complex numbers $\{z_n\}_{n=0}^{\infty}$ is computable if it is uniformly computable; that is, if there is an algorithm that given n, k as input produces a rational point q such that $|q - z_n| < 2^{-k}$.

A real a is *left-c.e.* if its left Dedekind cut is computably enumerable.

We now turn to computability of functions. First of all, by a *rational interval* we mean an interval whose endpoints are rational, and by a *rational rectangle* we mean a cartesian product of two open rational intervals. Suppose f is a function that maps complex numbers to complex numbers. We say that f is *computable* if there is an algorithm P that satisfies the following two criteria.

- **Approximation:** Whenever P is given an open rational rectangle R_1 as input, it either does not halt or produces an open rational rectangle R_2 such that $f[R_1] \subseteq R_2$.
- **Convergence:** If V is a neighborhood of $f(z)$, then z belongs to an open rational rectangle R_1 such that on input R_1 P produces an open rational rectangle $R_2 \subseteq V$.

These properties are illustrated in Figures 4.1 and 4.2. Less formally, the algorithm P computes arbitrarily good approximations of values of f from sufficiently good approximations of the corresponding arguments.

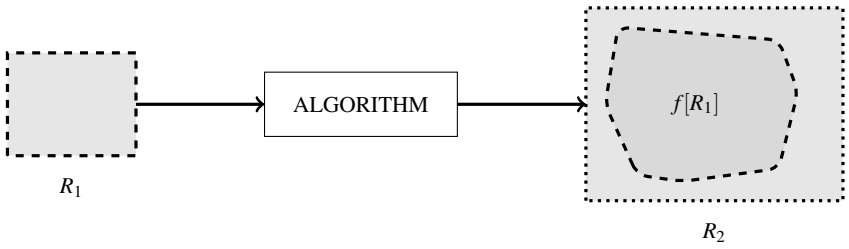


Fig. 4.1 The approximation property.

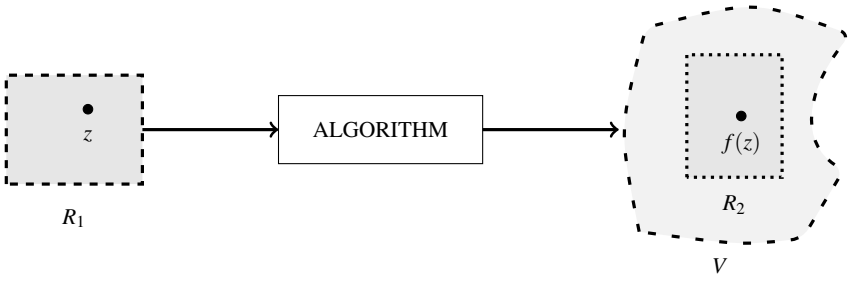


Fig. 4.2 The convergence property.

When we say that the algorithm is given a rational open rectangle as input, we are of course being a little sloppy. We do not literally mean that the entire set of points constituting the rational rectangle is provided as input. Instead, we mean that

some finite representation of the rectangle, such as the quadruple of its vertices, is provided as input and that such a finite representation is generated as output.

According to this definition, most functions one would expect to be computable, and that are commonly computed in practice, are in fact computable. For example, sin, cos, and exp are computable as can be seen by considering their power series expansions and the bounds on the convergence of these series that can be obtained from Taylor's Theorem.

Instead of open rational rectangles, we could have used open rational disks; the resulting class of computable functions would be the same. The rationale for this choice is that disks provide additional intrinsic information in the sense that when we say a point is not in a disk, we know that the distance from the point to the center of the disk is at least equal to the radius of the disk.

We now turn our attention to computability of sets. We define an open set $U \subseteq \mathbb{C}$ to be *c.e. open* if the set of all rational rectangles R such that $\overline{R} \subseteq U$ is computably enumerable. We define a closed set $C \subseteq \mathbb{C}$ to be *c.e. closed* if the set of all rational rectangles that contain a point of C is computably enumerable. We define a non-empty compact set $K \subseteq \mathbb{C}$ to be *computably compact* if there is an effective enumeration of the set of all finite collections $\{R_1, \ldots, R_k\}$ of rational rectangles such that $K \subseteq \bigcup_j R_j$ and $K \cap R_j \neq \emptyset$ for each j.

We can extend these ideas to the extended complex plane $\mathbb{C}_\infty = \mathbb{C} \cup \{\infty\}$ as follows. Define a *rational neighborhood of* ∞ to be a set of the form $\mathbb{C} - \overline{D(0;r)}$ where r is a positive rational number. Then, define a *rational neighborhood* to be a set that is either a rational rectangle or a rational neighborhood of infinity. We can then define computability of a function $f :\subseteq \mathbb{C}_\infty \to \mathbb{C}_\infty$ by taking the above definition and replacing 'rational rectangle' with 'rational neighborhood'. We can define c.e. open and c.e. closed subsets of $\mathbb{C}_\infty$ analogously.

If γ is a computable Jordan curve, then it follows from the constructive work of Gordon, Julian, Mines, and Richman that the interior and exterior of γ are c.e. open [34].

When a theorem of analysis is known not to be effectively true, attention then turns to pinning down the exact amount of extra information required to compute its solution operator. To be more precise, a typical theorem of analysis (or even other areas of mathematics) has the form

> If $a, b, c, \ldots$ satisfy …, then there is an A such that $A, a, b, c, \ldots$ satisfies ….

The effective version of such a statement has the form

> If $a, b, c, \ldots$ are computable and satisfy …, then there is a computable A such that $A, a, b, c, \ldots$ satisfies ….

An extra parameter $P_{a,b,c,\ldots}$ is *sufficient* if

> If $a, b, c, \ldots$ and $P_{a,b,c,\ldots}$ are computable, and if $a, b, c \ldots$ satisfy …, then there is a computable A such that $A, a, b, c, \ldots$ satisfies ….

It is *necessary* if there is a reversal:

If $a,b,c,\ldots$ and A are computable, and if $A,a,b,c\ldots$ satisfy $\ldots$ while $a,b,c\ldots$ satisfy $\ldots$, then $P_{a,b,c,\ldots}$ is computable.

In the remainder of this section we discuss computability of differentiation, series expansions, zeros, and the Open Mapping Theorem.

4.3.1 Computability of Differentiation, Computably Analytic Functions

As the following result of Myhill makes precise, differentiation is not a computable operator.

Theorem 4.3.1 ([71]). *There is a computable and continuously differentiable real-valued function f on the unit interval whose derivative is incomputable.*

However, differentiation of analytic functions with c.e. open domains is a computable operator.

Theorem 4.3.2 ([41]). *If f is a computable analytic function, and if the domain of f is c.e. open, then f' is computable.*

Furthermore, a code of a Turing machine that computes f' can be computed from a code of a Turing machine that computes f.

Accordingly, we then define an analytic function $f :\subseteq \mathbb{C} \to \mathbb{C}$ to be *computably analytic* if it is computable and if its domain is a c.e. open set. Note that we do not have to specify any computability conditions on the derivative of f.

The proof of Theorem 4.3.2 turns on the observation that integration is a computable operator (use upper and lower Riemann sums to get convergent upper and lower estimates on the integral) and the ability of the Cauchy Integral Formula to express differentiation of analytic functions in terms of integration.

4.3.2 Series

The following is an immediate consequence of Theorem 4.3.2.

Proposition 4.3.3. *If f is a computably analytic function, and if z_0 is a computable point in the domain of f, then the Taylor series expansion of f at z_0 can be computed.*

Since every analytic function has a valid Taylor series expansion, we tend to think of them as being completely described by power series. This works in the realm of classical mathematics, but in the domain of effective mathematics the series description of a function may fall quite short of providing the information necessary to compute the function, as is illustrated by the following.

Theorem 4.3.4 ([81]). *There is a computable sequence $\{a_n\}_{n=0}^{\infty}$ of complex numbers such that the series $\sum_{n=0}^{\infty} a_n z^n$ converges everywhere but defines an incomputable function.*

However, the function defined by the power series in Theorem 4.3.4 is at least 'locally computable'.

Theorem 4.3.5 ([81]). *If $\{a_n\}_{n\in\mathbb{N}}$ is a computable sequence of complex numbers, and if R is a computable real such that $\sum_{n=0}^{\infty} a_n z^n$ converges whenever $|z| < R$, then $\sum_{n=0}^{\infty} a_n z_0^n$ is computable whenever z_0 is a computable point such that $|z_0| < R$.*

The proof of Theorem 4.3.5 is a modification of the classical proof that a power series converges uniformly on compact subsets of its disk of convergence. Thus, it actually yields that if $\{a_n\}_{n=0}^{\infty}$ is a computable sequence of complex numbers, and if $r > 0$ is a computable real that is smaller than the radius of convergence of $f(z) = \sum_{n=0}^{\infty} a_n z^n$, then f is computable on $\overline{D(0;r)}$. In light of Theorem 4.3.4, the proof is necessarily non-uniform.

4.3.3 Zeros

In general, it is not possible to compute the zeros of a computable function, as illustrated by the following result of E. Specker.

Theorem 4.3.6 ([97]). *If $0 < \varepsilon < 1$, then there is a computable function $f : [0,1] \to \mathbb{R}$ that has no computable zeros and such that the Lebesgue measure of its zero set is larger than ε.*

However, since the zeros of a non-constant analytic function are isolated, we have the following.

Proposition 4.3.7 (Folklore). *Every zero of a computably analytic function is computable.*

Moreover, it follows from the Argument Principle that $\overline{f^{-1}[\{0\}]}$ is a c.e. closed set whenever f is a non-constant computably analytic function. That is, it is possible to enumerate the rational rectangles that contain a zero of f. In general, it is only possible to enumerate the rational open rectangles that do *not* contain a zero of a computable function.

If $f : D \to \mathbb{C}$ is analytic, and if $\{a_n\}_{n=0}^{\infty}$ is a sequence of points in D, then let us call $\{a_n\}_{n=0}^{\infty}$ a *primary zero sequence* of f if its terms are precisely the zeros of f, and the number of times each zero of f is repeated is its multiplicity. Again, by means of the Argument Principle we have the following.

Theorem 4.3.8 ([64]). *If f is a non-constant computably analytic function with infinitely many zeros, then it has a computable primary zero sequence.*

Furthermore, the proof of Theorem 4.3.8 is uniform.

4.3.4 Open Mapping

The following effective version of the Open Mapping Theorem for analytic functions is useful when studying computable conformal maps.

Theorem 4.3.9 (Effective Open Mapping Theorem [41]). *If f is a computably analytic function, and if U is a c.e. open subset of its domain, then $f[U]$ is c.e. open.*

Again, the proof is uniform in that a code of a Turing machine that computes an enumeration of the closed rational rectangles included in $f[U]$ can be computed from codes of Turing machines that compute f and an enumeration of the closed rational rectangles included in U.

M. Ziegler presents results on computable open mappings in real analysis in [103].

4.4 Computable Conformal Mapping of Simply Connected Domains

4.4.1 Classical Background

The history of the proofs of the Riemann Mapping Theorem provide part of the motivation for analyzing its effective content. We begin with the modern statement of Bernhard Riemann's famous theorem published in his *Inauguraldissertation* in 1851.

Theorem 4.4.1 (Riemann Mapping Theorem). *Suppose U is a simply connected domain whose boundary contains at least two points. Then, for every $z_0 \in U$, there is a unique conformal map ϕ of U onto the unit disk such that $\phi(z_0) = 0$ and $\phi'(z_0) > 0$.*

The map ϕ is called a *uniformizing* map or a *Riemann* map.

According to Gauss, who was the reviewer for the dissertation, "The whole is a solid work of high quality, not merely fulfilling the requirements usually set for a doctoral thesis, but far surpassing them" ([85], p. 183). Riemann's original formulation can be translated as follows: "Two given simply connected plane surfaces can always be mapped onto one another in such a way that each point of the one corresponds to a unique point of the other in a continuous way and the correspondence is conformal; moreover, the correspondence between an arbitrary interior point and an arbitrary boundary point of the one and the other may be given arbitrarily, but when this is done the correspondence is determined completely" [35]. Riemann sketched a proof for bounded domains with piecewise smooth boundaries based on the Dirichlet boundary value problem, which he solved by means of the Dirichlet principle. In 1870 Karl Weierstrass [100] constructed examples of simply connected domains for which the Dirichlet principle is not valid. Later David Hilbert [42, 43] provided a rigorous proof of the Dirichlet principle for the domains in Riemann's work.

Numerous mathematicians sought to find a correct proof, in the process developing new methods. Elwin Christoffel and Herman Schwarz developed integral formulas for the conformal transformation of the upper half-plane or the unit disk onto the interior of a polygon. Schwarz [94] and Carl Neumann [73] developed the alternating method, which allows one to solve the Dirichlet problem for domains which are the union of domains for which the Dirichlet problem can be solved. In 1887 A. Harnack [38] developed a different approach based on Green's function, a special kind of harmonic function which will be discussed in Section 4.6. H. Poincaré proved the uniqueness of the mapping function [78].

In 1900 W. F. Osgood [74] established the existence of Green's function for every simply connected domain. If $G(z,\zeta)$ is Green's function for U and if $p(z,\zeta)$ denotes its analytic extension, then $f_\zeta(z) = \exp p(z,\zeta)(z-\zeta)$ is the Riemann mapping of U onto the unit disk that maps ζ to the origin 0. Thus the Riemann mapping theorem follows from Osgood's work [99], but it "did not attract the attention it deserves" [1].

H. Poincaré [77] and, independently, F. Klein [48] conjectured the uniformization theorem for the Riemann surfaces of algebraic curves. The uniformization theorem states that every simply connected Riemann surface is conformally equivalent to one of the three Riemann surfaces: the open unit disk, the complex plane or the Riemann sphere. The first rigorous proofs of the uniformization theorem were given in 1907 by Poincaré [79] and, independently, by P. Koebe [51, 50, 49].

Constantine Carathéodory [16] gave the first purely function-theoretic proof of Riemann's theorem in 1912. He proved the uniqueness of the mapping function, a theorem due to Poincaré, using Schwarz's lemma alone (Carathéodory was the first to point out and name it). Carathéodory demonstrates the existence of the Riemann mapping by proving the convergence of conformal mappings of a sequence of domains. In a subsequent paper he proved that if U is a Jordan domain, then the Riemann mapping can be extended continuously to the boundary such that the extended function is a homeomorphism of $\bar{U}$ onto $\bar{\mathbb{D}}$ [17].

The first proof of Riemann's theorem that gave some idea as to how to build the uniformizing map was given by Paul Koebe [53], who called his method *Schmiegungsverfahren* (*osculation method*). As this method is the basis for much of the later work in constructive and computable analysis we will give a detailed description of it in Subsection 4.4.2.

In 1922 L. Fejér and F. Riesz demonstrated that the Riemann mapping function can be obtained as the solution of an extremal problem for the derivative. Their proof was presented by T. Radó in [83].

4.4.2 Computability Results

The utility of conformal maps and the non-constructive nature of many of these proofs leads to the question as to whether there is a truly constructive (in the spirit of Bishop), or at least effective, proof of the Riemann Mapping Theorem. At first

glance, it would appear that there is not. For, suppose a is a positive left-c.e. and incomputable real. Let D denote the disk with center 0 and radius a. Then, D is c.e. open. However, there is no computable conformal map of the unit disk onto D. For, suppose ϕ were such a map. Then, $\psi = \frac{1}{a}\phi$ is a conformal automorphism of the unit disk. So, ψ has the form

$$\psi(z) = \lambda \frac{z - z_0}{1 - z\overline{z_0}}$$

where $|z_0| < 1$ and $|\lambda| = 1$. It follows that z_0 is computable, and it then follows that a is computable; a contradiction.

What this example (due to Pour-El and Richards [82]) shows is that the computability of the domain U is insufficient information for the computation of a conformal map of U onto the unit disk. The question then becomes: what is a necessary and sufficient parameter for the computation of a Riemann map of a non-degenerate and simply connected domain? In 1999, P. Hertling showed that the answer is knowledge of the boundary; that is, an enumeration of the rational rectangles that contain a point of the boundary.

The journey toward this conclusion began in 1912 when P. Koebe described an iterative method for constructing a conformal map of a non-degenerate simply connected domain onto the unit disk [52]. As this method is the basis for later constructive work we describe it here. We begin with some intuition. To start, let U be a non-degenerate simply connected domain. Without loss of generality, we can assume $U \subseteq \mathbb{D}$. We then make the simple observation that if the square root function is iterated on a positive real, the results tend to 1. So, naively, we wish to iterate the square root function on U so that the boundary of U will be gradually pulled out to the boundary of the disk, and the sequence of iterates will thereby converge to a conformal map of U onto $\mathbb{D}$. The problem of course is that there is no complex square root function; there are merely analytic branches defined on simply connected domains that omit 0. In particular, if the sequence of iterates of U is to converge to the unit disk, then 0 must eventually belong to them. In fact, we can assume $0 \in U$. So, one might try to fix this flawed approach by first applying a conformal automorphism of the disk so as to move 0 out of the domain U and then applying a branch of square root for the resulting domain. But which conformal automorphism should be applied? The answer, it turns out, is to pick the one that moves the origin to a point on the boundary of U that is closest to the origin (among all points on the boundary of U).

To summarize, the steps in the Koebe construction are as follows. Suppose $0 \in U \subset \mathbb{D}$ is a non-degenerate and simply connected domain. Set $U_0 = U$. Suppose U_n has been defined such that $0 \in U_n \subset \mathbb{D}$. Let z_n be a boundary point of U_n such that

$$|z_n| = \min\{|z| \ : \ z \in \partial U_n\}.$$

Let:

$$\mu_1(z) = \frac{z - z_n}{1 - \overline{z_n}z}$$

$$\begin{aligned} s &= \text{analytic branch of } \sqrt{\ } \text{ on } \mu_1[U] \\ \mu_2(z) &= \frac{z - s(z_n)}{1 - \overline{s(z_n)}z} \\ f_n &= \mu_2 \circ s \circ \mu_1 \\ U_{n+1} &= f_n[U_n]. \end{aligned}$$

If we let $h_n = f_n \circ f_{n-1} \circ \ldots \circ f_0$, then the h_n's converge uniformly on compacta to a conformal map f. It also follows that $\lim_{n\to\infty} d(\partial U_n, \partial\mathbb{D}) = 1$, and so f maps U onto $\mathbb{D}$.

It is in general not possible to compute $z_n \in \partial U_n$ so as to minimize its distance to the origin. For, as first observed by E. Specker, although it is possible to compute the minimum and maximum value of a computable real-valued function on the unit interval, it is not in general possible to compute *where* these maxima and minima occur; specifically, there is a computable $f : [0,1] \to \mathbb{R}$ that does not achieve its maximum or minimum at any computable point [97]. The solution to this obstacle is that z_n need not be as close as possible to the origin, just 'close enough'. Specifically, it suffices to choose z_n such that

$$|z_n| < \frac{1}{2}(1 + d(\partial U_n, 0)).$$

If the boundary of U is a c.e. closed set, then it follows that the boundary of each U_n is a c.e. closed set, and so such a boundary point of U_{n+1} can indeed be computed. It is then possible to compute a modulus of uniform convergence. The key estimate is the following, which is Corollary 7.5 of Chapter 5 of [12].

Theorem 4.4.2. *Suppose U is an open subset of the plane and $D(0;r) \subseteq U$, where $0 < r < 1$. If $f : U \to \mathbb{C}$ is a conformal map such that $f(0) = 0$, $f'(0) > 0$, and $D(0;r) \subseteq f[U] \subseteq \mathbb{D}$, then*

$$|f(z) - z| \leq 3\frac{\sqrt{1-r}}{r^2 - |z|}$$

whenever $|z| < r^2$.

Via these techniques, Cheng [18] and Bishop and Bridges [12] proved the Riemann Mapping Theorem constructively, and Hertling proved the following.

Theorem 4.4.3 (Effective Riemann Mapping Theorem [41]). *Suppose U is a non-degenerate and simply connected domain, and let $z_0 \in U$. If z_0 is computable, and if U, ∂U are c.e. open and closed respectively, then the unique conformal map ϕ of U onto the unit disk such that $\phi'(z_0) > 0$ is computable.*

The proof of Theorem 4.4.3 is uniform. Theorem 4.4.3 establishes that the boundary of U provides *sufficient* information for computing a conformal map of U onto $\mathbb{D}$. By means of the Koebe $1/4$-estimate, Hertling also established the necessity of this parameter.

Theorem 4.4.4 (Effective Reverse Riemann Mapping Theorem [41]). *Suppose ϕ is a computable conformal map of a nondegenerate simply connected domain U onto the unit disk. Then, U is c.e. open and its boundary is c.e. closed.*

Again, the proof of Theorem 4.4.4 is uniform.

We note that prior to Hertling's work, Q. Zhou proved an effective version of the Riemann Mapping Theorem for so-called recursive open subsets of the plane [102] which imposes computability conditions on the set and its complement. The class of open sets considered by Hertling, which imposes computability conditions only on the set and its boundary, is strictly larger, and Theorem 4.4.4 shows it is the largest class of domains for which uniformizations can be computed.

4.4.3 Complexity Results

Theorems 4.4.3 and 4.4.4 pin down the exact information required to compute a Riemann map. However, the iterative method of the proof of Theorem 4.4.3 may converge very slowly in practice. We are thus led to consider complexity issues. That is, how quickly can the uniformizing map of a domain be computed? Theorems 4.4.3 and 4.4.4 demonstrate that knowledge of the boundary is essential for computation of these maps. So, it is natural to conjecture that the computational complexity of the boundary of a domain should determine the computational complexity of its uniformizing map.

The complexity of a computation is often influenced by the representation of its input data. In [10], I. Binder, M. Braverman, and M. Yampolsky consider two methods for representing the boundary of a bounded domain A:

1. A (code of a) Turing machine that computes a function $f : \mathbb{N}^3 \to \mathbb{N}$ such that $f(n,k,j) = 1$ whenever $D(k2^{-n} + ij2^{-n}; 2^{-n}) \cap \partial A \neq \emptyset$ and such that $f(n,k,j) = 0$ whenever $D(k2^{-n} + ij2^{-n}; 2 \cdot 2^{-n}) \cap \partial A = \emptyset$.
2. A set of $O(2^{2n})$ dyadic points S such that $d_H(S, \partial A) < 2^{-n}$ (where d_H is the Hausdorff distance).

In the first case, it is shown in the same paper that there is an algorithm that computes the values of a uniformizing map ϕ of D in time $2^{O(n)}$ with error at most 2^{-n} *provided* the *values* are sufficiently far from the boundary of D (specifically $|\phi(z)| < 1 - 2^{-n}$). R. Rettinger [87] showed that the lower bound is sharp. It is then shown that in the second case, the values of ϕ can be computed with error at most $O(2^{-n})$ in time polynomial in 2^n provided they are sufficiently far from the boundary (again, specifically $|\phi(z)| < 1 - 2^{-n}$). Fairly technical lower bound results are also demonstrated. Roughly speaking, in the first case it is shown that the complexity class **#P** is a lower bound, and in the second the circuit complexity class MAJ_n is a lower bound. It is notable that in the latter case the time complexity is quite high and is achieved via domains that are nevertheless fairly simple.

In terms of practical computation, one of the best-known algorithms is the Zipper algorithm, which was discovered independently by Kühnau [55] and Marshall

in the early 1980s. Similarly to the second algorithm considered by Binder, Braverman, and Yampolsky, it assumes the input data are a sequence of pairwise tangent disks that cover the boundary of D. Different versions of the algorithm were presented and convergence was discussed in [61]. The algorithm was discovered as an approximate method for conformal welding; it can also be viewed as a discretization of the Loewner differential equation. As noted in [10], the worst-case running time of Zipper exceeds that of the algorithm produced by Binder, Braverman, and Yampolsky. However, worst-case running time is not always the best method for comparing algorithms.

4.5 Boundary Extensions

When a conformal map ϕ maps a domain D_1 onto D_2, it is natural to inquire whether it extends to a continuous or even homeomorphic map of $\overline{D_1}$ onto $\overline{D_2}$. Such an extension, if it exists, is referred to as a *boundary extension*. If ϕ is computable, it is also natural to ask whether its boundary extension is computable. We will survey the classical material on the first question and results on the second.

4.5.1 Classical Background

Conformal maps do not always have boundary extensions. For example, no conformal map of the unit disk onto the domain in Figure 4.3 (obtained by removing all line segments of the form $\{2^{-n}\} \times [0, \frac{1}{2}]$ from the interior of the unit square) has a boundary extension.

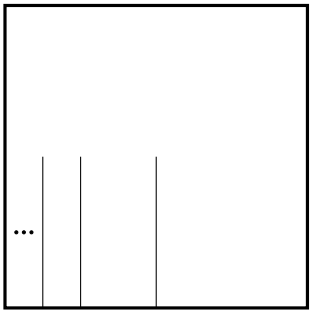


Fig. 4.3 A fine-tooth comb domain.

However, boundary extensions do exist if each domain is a Jordan domain.

Theorem 4.5.1 (Carathéodory Theorem). *If D is a bounded and simply connected Jordan domain, and if ϕ is a conformal map of D onto the unit disk, then ϕ extends to a homeomorphism of $\overline{D}$ with the closed unit disk.*

Recall that a topological space X is *locally connected* if for every $p \in X$ and every neighborhood U of p there is a connected open subset C of X such that $p \in C \subseteq U$. If X is a continuum, this is equivalent to the following assertion: for every $\varepsilon > 0$ there is a $\delta > 0$ such that whenever $p \in X$ there is a connected subset C of X such that $D(p;\delta) \subseteq C \subseteq D(p;\varepsilon)$. This condition is known as *uniform local connectivity*.

Theorem 4.5.2 (Generalized Carathéodory Theorem). *If ϕ is a conformal map of a bounded domain D onto the unit disk, and if the boundary of D is locally connected, then ϕ extends to a continuous map of $\overline{D}$ onto the closed unit disk. Conversely, if a conformal map of D onto the unit disk has a boundary extension, then the boundary of D is locally connected.*

The converse is an immediate consequence of the Hahn-Mazurkiewicz Theorem [44].

4.5.2 Computability Results

From our point of view, the natural questions to ask are whether boundary extensions of computable conformal maps are computable, and if not then what is a sufficient and necessary parameter for their computation? As it turns out, there is a computable conformal map (onto a Jordan domain) whose boundary extension is incomputable [102, 65]. So, the question turns to additional parameters. The following provides an answer for Jordan domains.

Theorem 4.5.3 (Effective Carathéodory Theorem [65]). *If γ is a computably parameterized Jordan curve, and if ϕ is a computable conformal map of the interior of γ onto the unit disk, then the boundary extension of ϕ is computable. Furthermore, if ψ is a computable conformal map of the interior of a Jordan domain D onto the unit disk, and if the boundary extension of ψ is computable, then the boundary of D has a computable parameterization.*

Thus, Theorem 4.5.3 says that in the case of a simply connected Jordan domain, a parameterization of the boundary provides both sufficient and necessary information for the computation of the boundary extension.

We note that having a computable parameterization of a Jordan curve γ is much more powerful than its computability as a closed or even as a compact set. For it follows essentially from results of J. Miller that there is a computably compact Jordan curve with no computable parameterization [70]; see also [102, 25]. While there is a computable conformal map of the unit disk onto the interior of such a curve, no such map has a computable boundary extension.

When $X \subseteq \mathbb{C}$, a *modulus of local connectivity* for X is a function $h : \mathbb{N} \to \mathbb{N}$ such that whenever $k \in \mathbb{N}$ and p, q are distinct points of X such that $|p - q| \leq 2^{-h(k)}$, X contains a path from p to q whose diameter is smaller than 2^{-k}. It follows that a compact and connected subset of the plane has a modulus of local connectivity if and only if it is locally connected. (See, e.g., Theorem 3-13 of [44]; the proof is essentially an application of the Lebesgue Number Theorem.) The essence of the proof of Theorem 4.5.3 is to compute a modulus of local connectivity for γ from a parametrization of γ, and then to use a length-area estimate to compute the boundary extension. In fact, Theorem 4.5.3 follows from the following quantitative version of Carathéodory's Theorem. When D is a Jordan domain and $\zeta_0 \in \partial D$, let $C(D; \zeta_0, r)$ denote the connected component C of $D(\zeta_0; r) \cap D$ such that $\zeta_0 \in \partial C$.

Theorem 4.5.4 ([69]). *Suppose ϕ is a conformal map of a Jordan domain D onto the unit disk. Let ζ_0 be a boundary point of D, and let $\varepsilon > 0$. Then, the diameter of $\phi[C(D; \zeta_0, r_0)]$ is smaller than ε whenever r_0 is a positive number that is smaller than*

$$\sup_{0<l<\varepsilon} \left(\exp\left(\frac{8\pi^2}{l^2 - \varepsilon^2} \right) \min \left\{ |\zeta_0 - \phi^{-1}(w)| \ : \ |w| \leq \sqrt{(1-l)^2 + \frac{\varepsilon^2 - l^2}{4}} \right\} \right). \tag{4.1}$$

When $0 < \varepsilon < 1$ and $l = \frac{\varepsilon}{2}$,

$$\frac{7}{16} < (1-l)^2 + \frac{\varepsilon^2 - l^2}{4} < 1.$$

Thus, (4.1) is positive when $0 < \varepsilon < 1$. In other words, for all sufficiently small $\varepsilon > 0$, there *is* a positive number r_0 that is smaller than (4.1). Such a number can be found via a search procedure.

From Theorem 4.5.4 we can also infer that the boundary extension of a computable conformal map of the disk onto a Jordan domain D can be computed from a modulus of local connectivity for the boundary of D. It is not difficult to show that a modulus of local connectivity for a Jordan curve can be computed from any one of its parameterizations. Thus, in the case of a Jordan domain, a modulus of local connectivity for the boundary also provides a necessary and sufficient parameter for the computation of its boundary extension.

More about computable moduli of local connectivity, other notions of effective local connectivity, and applications to space-filling curves can be found in [25] and [21].

We now turn to the computation of boundary extensions of conformal maps onto non-Jordan domains. Suppose D is a c.e. non-degenerate and simply connected domain. Assume also that the boundary of D is locally connected. As has been noted, it is quite possible that no conformal map of the unit disk onto D has a computable boundary extension. So, once again the problem is to pin down a sufficient and necessary parameter for their computation. In light of our discussion of Jordan domains, one might expect that a modulus of local connectivity for the boundary provides the right amount of extra information. Indeed, it provides a sufficient amount.

Theorem 4.5.5 (Effective Boundary Extension Theorem [68]). *If ϕ is a computable conformal map of the unit disk onto a domain D whose boundary has a computable modulus of local connectivity, then the boundary extension of ϕ is computable.*

The key estimate in the proof is the following.

Lemma 4.5.6. *Suppose ζ, r_0, r_1, α_1, α_2, Ω are as in Figure 4.4. That is:*

1. $0 < r_0 < r_1 < 1$, *and* $|\zeta| = 1$.
2. α_1 *and* α_2 *are disjoint crosscuts of*

$$\{z \in \mathbb{D} \,:\, r_0 < |z - \zeta| < r_1\}$$

   that do not touch the boundary of $\mathbb{D}$.
3. Ω *consists of those points in the side of* α_1 *that includes* α_2 *that also belong to the side of* α_2 *that includes* α_1.

Then,

$$Area(\phi[\Omega]) \geq \frac{1}{\pi} d_{inf}(\phi[\alpha_1], \phi[\alpha_2])^2 \ln\left(\frac{r_1}{r_0}\right).$$

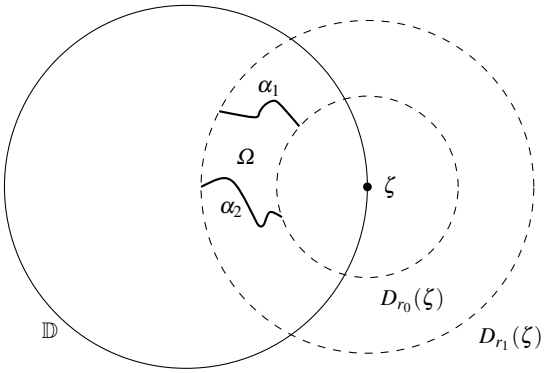


Fig. 4.4 Assumptions in Lemma 4.5.6.

However, in [63], it is shown that there is a computable conformal map ϕ of the unit disk onto a domain D whose boundary extension is computable even though the boundary of D does not have a computable modulus of local connectivity. Thus a modulus of local connectivity does not provide a necessary extra parameter for the computation of boundary extensions. However, by utilizing Wolff's estimate (see [80]), it can be shown fairly easily that a modulus of local connectivity for the *complement* of D provides a sufficient extra parameter. That is, if ϕ is a computable conformal map of the unit disk onto a domain D, and if $\mathbb{C} - D$ has a computable modulus of local connectivity, then the boundary extension of ϕ is computable. By

means of the estimate in Lemma 4.5.6, it can also be shown that a modulus of local connectivity for the complement is a necessary extra parameter. That is, if ϕ is a conformal map of the unit disk onto a domain D with a locally connected boundary, and if the boundary extension of ϕ is computable, then the complement of D has a computable modulus of local connectivity.

I. Binder, C. Rojas, and M. Yampolsky [11] obtained a necessary and sufficient condition for the computability of ϕ using the *Carathéodory modulus*. They also show that the Carathéodory modulus cannot be replaced by the modulus of local connectivity. These results are presented in Chapter 5 of this volume.

4.6 Harmonic Functions

Before we consider conformal mapping of multiply connected domains, we will first cover computability of harmonic functions; in particular conjugates, analytic extensions, and Dirichlet problems. This is because the computation of these maps can in some cases be reduced to the computation of solutions to Dirichlet problems.

4.6.1 Classical Background

4.6.1.1 Conjugates and Analytic Extensions

Suppose u is a harmonic function with domain D. If $\overline{D(0;r)} \subseteq D$, then a harmonic conjugate of u on $D(0;r)$ is given by

$$\tilde{u}(z) = \int_0^{2\pi} u(re^{i\theta}) \frac{re^{-i\theta}z - re^{i\theta}\overline{z}}{|re^{i\theta} - z|^2} \mathrm{d}\theta.$$

So, every harmonic function has a conjugate at least locally. If D is simply connected, then a harmonic conjugate v of u is given by

$$v(\zeta) = \int_{\zeta_1}^{\zeta} \frac{\partial u}{\partial n} \mathrm{d}s \tag{4.2}$$

where ζ_1 is a fixed point in D, $\frac{\partial u}{\partial n}$ denotes the normal derivative of u (that is, the derivative of u in the direction that is normal to the path of integration), and $\mathrm{d}s$ is the differential of arc length. Specifically, if the path of integration is a smooth curve given by $\gamma(t) = x(t) + iy(t)$, then

$$\frac{\partial u}{\partial n}(t) = \left(\frac{\partial u}{\partial x} y'(t) - \frac{\partial u}{\partial y} x'(t) \right) \frac{1}{|x'(t) + iy'(t)|}$$

and $\mathrm{d}s = |x' + iy'|\mathrm{d}t$. The Cauchy-Riemann equations can thus be rewritten as the single equation $\frac{\partial u}{\partial n} = \frac{\partial v}{\partial s}$.

If D is multiply connected, then the right-hand side of (4.2) may depend on the path of integration. In this case, u is said to have a *multi-valued harmonic conjugate*. Suppose D is a bounded domain and that its boundary consists of n Jordan curves $\gamma_1, \ldots, \gamma_n$ with γ_n being the outermost. The *period of the conjugate of u around* γ_j is defined to be

$$\frac{1}{2\pi}\int_\gamma \frac{\partial u}{\partial n}\mathrm{d}s$$

where $\gamma \subseteq \mathrm{dom}(u)$ is homotopic to γ_j. It follows that u has a (single-valued) harmonic conjugate if and only if its conjugate periods around $\gamma_1, \ldots, \gamma_{n-1}$ are all 0 (for then, the path of integration in (4.2) makes no difference).

If u does not have a conjugate, then there is a well-known procedure for 'extending' u so that it does; we describe it now. Let P_j denote the conjugate period of u about γ_j. It follows from the results discussed below on Dirichlet problems that for each j there is a harmonic function $\omega(\cdot, \gamma_j, D)$ on D such that for each $\zeta \in \partial D$,

$$\lim_{z\to\zeta} \omega(z, \gamma_j, D) = \begin{cases} 1 & z \in \gamma_j, \\ 0 & z \notin \gamma_j. \end{cases}$$

The function $\omega(\cdot, \gamma_j, D)$ is called the *harmonic measure function* of γ_j. When context makes other parameters clear we abbreviate $\omega(\cdot, \gamma_j, D)$ by ω_j. More generally, a harmonic measure function $\omega(\cdot, E, D)$ exists for each Borel set $E \subseteq \partial D$; see, e.g., Chapter III of [32]. Let $P^D_{k,j}$ be the conjugate period of ω_j about γ_k. It is well known that the matrix $(P^D_{k,j})_{k,j=1,\ldots,n-1}$ (known as the *Riemann matrix*) is invertible. Thus, there exist unique $b_1, \ldots, b_{n-1}$ such that

$$\sum_{j=1}^{n-1} P^D_{k,j} b_j = -P_k.$$

It follows that

$$u + \sum_{j=1}^{n-1} b_j \omega_j$$

has period 0 around each of $\gamma_1, \ldots, \gamma_{n-1}$. Thus, it has a harmonic conjugate, which we denote by $\tilde{u}$. We let $\hat{u} = u + i\tilde{u}$ and refer to $\hat{u}$ as an *analytic extension* of u.

4.6.1.2 Dirichlet Problems

Suppose we are given a bounded domain $D \subseteq \mathbb{C}$ and a piecewise continuous function $f : \partial D \to \mathbb{R}$. The resulting *Dirichlet problem* is to find a harmonic function on D, u, such that

$$\lim_{z\to\zeta} u(z) = f(\zeta) \tag{4.3}$$

for all $\zeta \in \partial D$ at which f is continuous. It is well known that this Dirichlet problem has a solution when the boundary of D consists of finitely many pairwise disjoint Jordan curves. It then follows from the Generalized Carathéodory Theorem (Theorem 4.5.2) that this Dirichlet problem has a solution when the boundary of D is locally connected. Once the existence of u is demonstrated, uniqueness follows immediately from the Maximum Principle for harmonic functions. That is, there is exactly one harmonic function on D, u, for which Equation (4.3) holds. See, e.g., Chapters I and II of [32]. Accordingly, we denote this function u_f. Dirichlet problems for other kinds of domains exist but will not be considered here.

When D is simply connected, that is when D is the interior of a Jordan curve, it is fairly straightforward to prove the existence of u_f. In particular, when D is the unit disk $\mathbb{D}$,

$$u_f(z) = \frac{1}{2\pi} \int_{\partial \mathbb{D}} f(\zeta) P(z, \zeta) \mathrm{d}s_\zeta, \tag{4.4}$$

where

$$P(z, \zeta) = \frac{1 - |z|^2}{|z - \zeta|^2}$$

and $\mathrm{d}s_\zeta$ is the differential of the arc length with respect to the variable ζ. The function P is called the *Poisson kernel*, and Equation (4.4) is known as the *Poisson Integral Formula*. Since the composition of a harmonic function with an analytic function yields a harmonic function, the existence of u_f for simply connected Jordan domains now follows from the Riemann Mapping Theorem and the Carathéodory Theorem. It then also follows for simply connected bounded domains with locally connected boundary.

When D is bounded by more than one Jordan curve, there are at least two methods available to prove the existence of u_f. One is an extreme-value argument as in Section 6.4.2 of [1]. If one desires a constructive proof, a natural choice is the Schwarz alternating method which is described in Section IV.2 of [23] and in Chapter II of [32]. A sequence of harmonic functions $u_1 \geq u_2 \geq u_3 \ldots$ that converges to u is produced thereby. However, the proof does not give any information about the rate of convergence, which is essential for computability considerations. In [67], this approach is reworked more carefully so as to give an explicit rate of convergence. Moreover, it is shown that the Schwarz alternating method can be recast as an iterative construction so that convergence is proved via the Contraction Mapping Theorem and yields a closed form for the solution of the form

$$u_f(z) = h(z) + \int_{B_2} \mathbf{K}(z, \zeta_1) h(\phi_2(\zeta_1)) \mathrm{d}s_{\zeta_1}.$$

The kernel $\mathbf{K}$ is obtained through kernel iteration

A harmonic function of particular interest is Green's function, which is constructed as follows. Suppose D is a bounded domain whose boundary consists of finitely many pairwise disjoint Jordan curves. For the moment, fix a point $\zeta_1 \in D$. Let $h(\zeta) = -\log|\zeta - \zeta_1|$ for all ζ on the boundary of D. Then, let $G_D(z; \zeta) = u_h(z) + \log|z - \zeta_1|$. G_D is known as Green's function. One of its important proper-

ties is that it yields a closed form for the solutions of the Dirichlet problem. Namely,

$$u_f(z) = \frac{1}{2\pi}\int_{\partial D} \frac{\partial G(z;\zeta)}{\partial n_\zeta} f(\zeta)\mathrm{d}s_\zeta.$$

(See Chapter II of [32].)

4.6.1.3 Neumann Problems

Let D be a bounded domain with smooth boundary curves $\gamma_1, \ldots, \gamma_n$. Let f be a continuous function on the boundary of D, and suppose $\int_{\partial D} f\mathrm{d}s = 0$. The resulting *Neumann problem* is to find a harmonic function u on D such that

$$\frac{\partial u}{\partial n} = f \text{ on } \partial D, \tag{4.5}$$

$$\int_{\partial D} u \,\mathrm{d}s = 0. \tag{4.6}$$

Such solutions exist (see, e.g., Appendix B of [32]). Condition (4.6) ensures they are unique. We then obtain the existence of the *Neumann function* of D, N_D, which is defined by the following conditions.

1. $z \mapsto N_D(z,\zeta) + \log|z-\zeta|$ is harmonic.
2. $\frac{\partial}{\partial n_z} N_D(z,\zeta) = -\frac{2\pi}{L}$ on ∂D, where L is the length of ∂D.
3. $\int_{\partial D} N_D(z,\zeta)\mathrm{d}s_z = 0$.

4.6.2 Computability Results

The following is an immediate consequence of Theorem 4.3.2 and the local conjugate formula in Subsection 4.6.1.

Corollary 4.6.1 ([4]). *If u is a computably harmonic function, then u' is computable.*

We now discuss computability of conjugates, analytic extensions, and Dirichlet problems.

4.6.2.1 Conjugates and Analytic Extensions

Essentially by following the procedure outlined in Subsection 4.6.1, we obtain a proof of the following.

Theorem 4.6.2 ([6]). *Suppose u is a computable harmonic function defined on a bounded domain D, and suppose the boundary of D consists of computable and*

computably differentiable smooth Jordan curves $\gamma_1, \ldots, \gamma_n$ of which γ_n is the outermost. Then, there exist computable $b_1, \ldots, b_{n-1}$ such that

$$u + \sum_{j=1}^{n-1} b_j \omega(\cdot, \gamma_j, D) \tag{4.7}$$

has a harmonic conjugate.

Furthermore the proof of Theorem 4.6.2 is uniform. It follows that the analytic extension of u can also be computed.

4.6.2.2 Dirichlet Problems

We begin with simply connected domains.

Theorem 4.6.3 ([65]). *Suppose the unit circle is decomposed into computable arcs $\gamma_1, \ldots, \gamma_n$ that intersect only at their endpoints. Suppose that f_j is a computable real-valued function on γ_j for each j and that*

$$f(\zeta) = \begin{cases} f_j(\zeta) & \zeta \in \gamma_j, \zeta \neq \gamma_j(0), \gamma_j(1), \\ \max_j f_j(\zeta) & \textit{otherwise}, \end{cases}$$

whenever $|\zeta| = 1$. Then, the solution u_f of the resulting Dirichlet problem is computable. In addition, the extension of u_f to $\overline{\mathbb{D}}$ is computable except at the endpoints of the arcs $\gamma_1, \ldots, \gamma_n$.

When we say that the arcs $\gamma_1, \ldots, \gamma_n$ are computable, we mean that their endpoints are computable. Note that the computability of u_f follows from the Poisson Integral Formula (Equation 4.4). Thus, the more significant part of Theorem 4.6.3 is the computation of the boundary extension.

It now follows from the Effective Carathéodory Theorem and the Effective Boundary Extension Theorem that Dirichlet problems on simply connected Jordan domains and more generally simply connected domains with locally connected boundary are computable.

In [10], the harmonic measure function for simply connected domains is shown to be computable by means of a random walk argument based on a famous theorem of Kakutani that gives a probabilistic interpretation of harmonic measure [45] (see also Appendix F of [32]).

We now turn to multiply connected domains. The following can be proven via the iterative method in [67] that was discussed in Subsection 4.6.1.

Theorem 4.6.4 (Computable Solution of Dirichlet Problems [4]). *Suppose D is a bounded domain whose boundary consists of finitely many pairwise disjoint Jordan curves, and suppose f is a computable real-valued function on the boundary of D. Then, the solution u_f of the resulting Dirichlet problem is computable, as is its continuous extension to $\overline{D}$.*

Again, the proof of Theorem 4.6.4 is uniform.

The most general result has been proven by Binder, Braverman, Rojas, and Yampolsky [9]. An open and connected domain $\Omega \subset \hat{\mathbb{C}}$ is said to satisfy the *capacity density condition* if there exists a constant $C > 0$ such that

$$\mathrm{Cap}(B(x,r) \cap \partial\Omega) \geq Cr$$

for all $x \in \partial\Omega$ and $r \leq r_0$. Here $\mathrm{Cap}(\cdot)$ denotes the logarithmic capacity as defined in Section 2.3 of [9].

Theorem 4.6.5 (Computable Solution of Dirichlet Problems [9]). *Let $\Omega \subset \hat{\mathbb{C}}$ be the complement of a computable compact set K and x_0 be a point in Ω. Suppose Ω is connected and satisfies the capacity density condition. Then the harmonic measure ω_{Ω,x_0} is computable with an oracle for x_0.*

They also present an example (Section 5.2 of [9]) of a computable closed set with a non-computable harmonic measure of the complement, thus demonstrating that the computability of the set K is not enough to ensure the computability of the harmonic measure.

Theorem 4.6.5 is based (see Theorem 5.4 of [9]) on an earlier work of Binder and Braverman [8], where they study the complexity of the Dirichlet problem (Corollary 1 of [8]) by a discretization of the continuous Dirichlet problem.

4.6.2.3 Neumann Problems

We conclude this section with the computability of Neumann problems. The following was proven by Andreev and McNicholl in 2012 [6].

Theorem 4.6.6 (Computable solution of Neumann problems [6]). *Suppose D is a bounded domain whose boundary consists of finitely many pairwise disjoint computable and computably differentiable Jordan curves. Suppose also that f is a computable real-valued function on the boundary of D such that $\int_{\partial D} f \, \mathrm{d}s = 0$. Then, the solution of the resulting Neumann problem is computable.*

The proof of Theorem 4.6.6 is uniform.

4.7 Conformal Mapping of Multiply Connected Domains

4.7.1 Classical Background

When considering conformal mapping of domains that are not simply connected, it is convenient to move the domain of discourse to the extended complex plane. As in the complex plane, a set $D \subseteq \mathbb{C}_\infty$ is a domain if it is open and connected. Throughout

this section, all domains are domains in the extended complex plane. A domain is *n-connected* if its boundary has n connected components. Each of these components is a continuum, and so we say a domain is *degenerate* if one of the components of its boundary consists of a single point.

We note that if $D \subseteq \mathbb{C}$ is simply connected, then it is 1-connected as a domain in $\mathbb{C}_\infty$. But the class of 1-connected domains is larger than the class of simply connected domains. For example, $\mathbb{C}_\infty - \overline{\mathbb{D}}$ is 1-connected. By inversion, every 1-connected domain is conformally equivalent to a simply connected domain. We are thus led to the following reformulation of the Riemann Mapping Theorem.

Theorem 4.7.1 (Riemann Mapping Theorem for $\mathbb{C}_\infty$). *If D is a non-degenerate 1-connected domain, then there is a conformal map of D onto $\mathbb{D}$.*

For theoretical and practical reasons, the unit disk became the standard domain in the study of conformal mappings on 1-connected domains; hence it is also called the *canonical domain*. When we discuss conformal maps on multiply connected domains, we first ought to notice that since these maps are continuous, they will preserve the connectivity of the domains. Thus, we have to introduce canonical domains for each order of connectivity. But two non-degenerate n-connected domains are not necessarily conformally equivalent, as the following example from Nehari's text shows [72]. Let $A(c;r,R)$ denote the annulus with center c and inner and outer radii r and R respectively. Suppose $0 < r < R < 1$ and that f is a conformal mapping of the annulus $A_1 = A(0;r,1)$ onto the annulus $A_2 = A(0;R,1)$. Let $u(z) = \log r \log|f(z)| - \log R \log|z|$. Thus, u is harmonic on A_1, and by the Maximum Principle for harmonic functions, u is identically 0. However, if $r \neq R$, then the conjugate period of u is nonzero, and so u has no conjugate; a contradiction.

It also follows from this argument that two annuli $A(z_0;r_1,R_1)$ and $A(z_0;r_2,R_2)$ are conformally equivalent only if $R_1/r_1 = R_2/r_2$.

Any 2-connected domain A can be conformally mapped onto an annulus, and its conformal type is completely determined by the ratio of the inner and outer radii, which is called the modulus of A. The conformal type of a domain of connectivity $n > 2$ is determined by $3n-6$ real numbers called the *moduli*, or *Riemann moduli*, of the domain. To see how the number $3n-6$, $n > 2$, of moduli is obtained, first notice that rotation, shift, and dilation/contraction do not change the conformal type of a domain. Consider a domain obtained by removing $n-2$ disks from an annulus $A(0;r,R)$ with $r < 1 < R$. By rotation and dilation/contraction we can place the center of one of the disks at 1. Thus, this disk is described by one real parameter, its radius. Each of the remaining disks is described by three real parameters: two for the center and one for the radius. Hence, we need $3 + 3(n-3) = 3n-6$ real parameters to describe the conformal type.

As another example consider the domain obtained by removing $n-2$ concentric arcs from an annulus $A(z_0;r,R)$ with $r < 1 < R$. Of the two terminal points on the arc, let the initial point be the point for which the arc is in counterclockwise direction from this point. By rotation and dilation/contraction we can place the initial point of one arc at 1. Thus to describe this arc we need to know only its length. For all other arcs we need to know three real parameters: two for the initial point and one

for its length. Hence again we need $3n-6$ real parameters. For another example, see Nehari [72, Exercise 13, p. 354].

To bypass this cumbersome situation, the definition of a canonical domain is based on its geometric character rather than on the moduli. A *canonical domain class* is a class of domains with common geometric features such that every finitely connected domain is conformally equivalent to at least one domain in the class. With each canonical domain class $\mathscr{K}$, there is thus the associated problem of finding a conformal map of a finitely connected non-degenerate domain onto a domain in $\mathscr{K}$.

P. Koebe described 39 canonical domains with connectivity n, $n>2$, whose common geometric feature is that at least one boundary component is a circular arc, a segment, a ray, or a straight line [54]. These are referred to as *slit domains*. We will discuss these domains and classical techniques for constructing conformal maps onto them below.

One notable absence from Koebe's list are the circular domains. These are Jordan domains whose boundary curves are all circles. These are quite singular for several reasons. For one, whereas there are closed-form expressions for conformal mappings onto the slit domains, there is no such formula for conformal maps onto circular domains. In addition, while there are standard conformal maps between many slit domains, there are *no standard maps of slit domains onto circular domains*. These domains are the canonical domains in a number of recent studies of the Schwarz-Christoffel formula for multiply connected domains, nonlinear problems in mechanics, and in aircraft engineering. See, for example, [3, 27, 24].

We now describe some of the slit domains and the circular domains and classical work on their conformal mapping problems.

4.7.1.1 Slit Domains

A treatment of all 39 slit domains identified by Koebe is beyond the scope of this survey. We confine ourselves to the following.

- *Slit disk domains:* These domains are obtained by removing finitely many arcs from $\mathbb{D}$. Each of these arcs must be an arc of a circle centered at the origin.
- *Slit annulus domains:*These domains are obtained by removing from an annulus finitely many concentric circular arcs.
- *Circular slit domains:* These domains are obtained by removing from $\mathbb{C}_\infty$ one or more circular arcs each of which is centered at the origin.
- *Radial slit domains:* These domains are obtained by removing from $\mathbb{C}_\infty$ one or more line segments which do not pass through the origin. Each of these line segments, when extended indefinitely in both directions, must yield a line that passes through the origin.
- *Parallel slit domains:* These domains are obtained by removing from $\mathbb{C}_\infty$ one or more parallel line segments.

In an appendix to Courant's book [22], Max Schiffer [93] applied the theory of Green's function to boundary value problems for multiply connected domains of

the above type. We first discuss his construction of the slit disk map. Specifically, let D be a domain bounded by n smooth curves, which can always be achieved by preliminary transformations. Fix a point $\zeta_1 \in D$, and set

$$f(z) = \exp(\hat{G_D}(z;\zeta))(z-\zeta_1).$$

It follows that f maps D onto a circular domain. *The key point here is that the slit disk map can be explicitly reduced to Green's function for D.*

We now consider the radial slit domains. Let $u_1(z) = N_D(z,\zeta_0') - N_D(z,\zeta_1') + \log|z-\zeta_0'| - \log|z-\zeta_1'|$. (Recall from Section 4.6 that N_D is the Neumann function for D.) Hence, u_1 is harmonic in D. It also follows that each conjugate period of u_1 is 0, so u_1 has a single-valued harmonic conjugate v_1. Let $f_1 = u_1 + iv_1$, and let

$$g(z) = \exp(f_1(z))\frac{z-\zeta_1'}{z-\zeta_0'}.$$

It is shown in the proof of (A1.62) of [93] that g is the required conformal map of D_1 onto a radial slit domain.

Once the slit disk map and radial slit maps are known, many of the other slit domain maps can be obtained explicitly. We refer the reader to Schiffer's essay [93] or to [6] for details.

4.7.1.2 Circular Domains

In 1910, Koebe suggested a way to construct the Riemann mapping from a multiply connected planar domain to a circular domain of the same connectivity. Koebe's construction sidesteps many obstacles but the error estimates are based on data of the image domain (i.e., the circular domain), which is not known at the start of the process, rather than on data of the initial domain. Error estimates derived solely from the geometry of the initial domain were discovered almost a hundred years later by V. V. Andreev and T. H. McNicholl.

We describe Koebe's construction as follows. Let D be an unbounded non-degenerate n-connected domain. We define sequences $\{D_k\}_{k=0}^{\infty}$, $\{D_{k,1}\}_{k=0}^{\infty}$, ..., $\{D_{k,n}\}_{k=0}^{\infty}$, and $\{f_k\}_{k=0}^{\infty}$ inductively as follows. To begin, let $D_{0,1},\ldots,D_{0,n}$ be the connected components of $\mathbb{C}_\infty - D$. Let $D_0 = D$ and let $f_0 = Id_D$. Let $k \in \mathbb{N}$, and suppose $f_k, D_k, D_{k,1},\ldots,D_{k,n}$ have been defined. Let $k' \in \{1,\ldots,n\}$ be equivalent to $k+1$ modulo n. Define f_{k+1} to be the conformal map of $\hat{\mathbb{C}} - D_{k,k'}$ onto a 1-connected circular domain C such that $f_{k+1}(z) = z + O(z^{-1})$. We then set $D_{k+1} = f_{k+1}[D_k]$. Let $D_{k+1,j} = f_{k+1}[D_{k,j}]$ when $j \neq k'$, and let $D_{k+1,k'} = \hat{\mathbb{C}} - C$. Finally, let $g_k = f_k \circ \ldots \circ f_0$. It is well-known (and quite surprising!) that $\lim_{k\to\infty} g_k = f_D$.

D. Gaier proved the convergence of this method and also obtained an upper bound on the error in the Koebe construction which tends to zero as the iterations progress [30]. However, the constants in this estimate are obtained from parameters associated with the target circular domain, which is usually not known in advance (and if it were there would be much easier methods for calculating the circular do-

main map; just map the initial domain and the target circular domain to the same slit domain). In [40], Henrici presents a modification of Gaier's construction. But, again, Henrici's bounds use certain numbers associated with the circular domain C, which usually is not known in advance. So, a central problem of the Koebe construction is to find an error bound phrased entirely in terms of information about the input domain D. Such an estimate can be obtained as follows.

Suppose D is an unbounded circular domain with boundary curves $\Gamma_1,\ldots,\Gamma_n$, and let

$$\mathscr{E}_R = \min_j \exp\left\{-\left(\frac{P^D_{j,j}}{\omega(\infty,\Gamma_j,D)^2}+1\right)R^2\right\} \tag{4.8}$$

for all $R>0$. Fix $j\in\{1,\ldots,n\}$. We define a number λ_j^D as follows. Let g be a conformal map of the exterior of Γ_j onto the interior of Γ_j. Let $E_1 = g[D]$. Let E_2 be the unbounded domain whose boundary components are $g[\Gamma_k]$, where $k\in\{1,\ldots,n\}-\{j\}$. Hence, E_2 is an $(n-1)$-connected domain. Let h be a conformal map of E_2 onto an unbounded circular domain normalized so that $h(z) = z+O(z^{-1})$, and let $E_3 = h[E_1]$. Therefore, E_3 is bounded by the curves $h[\partial\Gamma_j]$, $hg[\Gamma_1],\ldots,hg[\Gamma_{j-1}],hg[\Gamma_{j+1}],\ldots,hg[\Gamma_n]$. The last $n-1$ of these curves are circles. Set

$$M = 2\max\{|z| \,:\, z\in\overline{E_3}\}.$$

Thus, $\frac{1}{M}E_3\subset\mathbb{D}$. Let $\mathscr{C}_1,\ldots,\mathscr{C}_{n-1}$ label the circles

$$\frac{1}{M}hg[\Gamma_1],\ldots,\frac{1}{M}hg[\Gamma_{j-1}],\frac{1}{M}hg[\Gamma_{j+1}],\ldots,\frac{1}{M}hg[\Gamma_n],$$

and set

$$\lambda_j^D = \min_{k_1\neq k_2}\rho(\mathscr{C}_{k_1},\mathscr{C}_{k_2}), \tag{4.9}$$

where ρ denotes the hyperbolic pseudometric on $\mathbb{D}$.

Recent results by Ransford and Rostand on the computation of capacity led to the following [84].

Theorem 4.7.2 ([5]). *Suppose D is an unbounded circular domain and $D_{2/R}(z_0)\subseteq\mathbb{C}_\infty - D\subseteq D_{R/2}(0)$. Let $g_0,g_1,\ldots$ denote the finite compositions generated by the Koebe construction, and let $f_D = \lim_j g_j$. Then, for all $j\in\mathbb{N}$, $\left\|g_j - f_D\right\|_\infty < \gamma_D^+(\mu_D^+)^{\lfloor j/n\rfloor}$, where:*

$$\gamma_D^+ = 72R\left[\frac{1}{\mathscr{E}_R^2\log(1+\frac{1}{4}\mathscr{E}_R^3\min\{\lambda_1,\lambda_2\})}+1\right]\frac{\mathscr{E}_R^2\min\{\lambda_1,\lambda_2\}+1}{\mathscr{E}_R^3\min\{\lambda_1,\lambda_2\}}$$

$$\mu_D^+ = \frac{1}{1+\frac{1}{4}\mathscr{E}_R^3\min\{\lambda_1,\lambda_2\}}.$$

Again, note that the estimates in Theorem 4.7.2 are obtained solely from geometric features of the starting domain D without prior knowledge of the corresponding circular domain.

We now turn to the computability results for these canonical domain classes.

4.7.2 Computability Results

4.7.2.1 Slit Domains

By means of the computability results on harmonic functions discussed in Section 4.6, we obtain the following two theorems.

Theorem 4.7.3 (Computability of slit disk mapping [6]). *Suppose D is a finitely connected and non-degenerate domain. Assume also that D is c.e. open and that $\zeta_0 \in D$ is computable.*

1. *If the boundary of D is c.e. closed, then there is a computable conformal map f of D onto a slit disk domain with computable boundary curves such that $f(\zeta_0) = 0$.*
2. *If there is a computable conformal map of f onto a slit disk domain with computable boundary curves, then the boundary of D is c.e. closed.*

Theorem 4.7.4 (Computability of radial slit mapping [6]). *Suppose D is a finitely connected and non-degenerate domain. Assume also that D is c.e. open and that $\zeta_0, \zeta_1 \in D$ are distinct computable points.*

1. *If the boundary of D is c.e. closed, then there is a computable conformal map f of D onto a radial slit domain with computable boundary curves such that $f(\zeta_0) = 0$ and $f(\zeta_1) = \infty$.*
2. *Conversely, if there is a computable conformal map of D onto a radial slit domain with computable boundary curves, then the boundary of D is c.e. closed.*

Both proofs are uniform.

4.7.2.2 Circular Domains

By the Effective Riemann Mapping Theorem, all of the steps in the Koebe construction can be carried out effectively. It follows from the results on computability of harmonic functions in Section 4.6 that the constants $\mathscr{E}_R$ in Equation 4.8 can be computed. If we assume by way of induction that the circular domain maps are computable for $(n-1)$-connected domains, then the constants λ_j^D defined in Equation 4.9 are computable as well. We thus have a uniform proof of the following.

Theorem 4.7.5 (Computability Circular Domain Map [4]). *Suppose D is a non-degenerate finitely connected domain. Assume that D is c.e. open and that $\zeta_0 \in D$ is computable.*

1. *If the boundary of D is c.e. closed, then there is a computable conformal map f of D onto an unbounded circular domain with computable boundary curves such that $f(\zeta_0) = \infty$.*

2. *Conversely, if there is a computable conformal map of D onto an unbounded circular domain with computable boundary curves, then the boundary of D is c.e. closed.*

We note that Theorem 4.7.5 can be proven non-uniformly by taking either Gaier or Henrici's bound on the error in the Koebe construction and replacing the constants therein with sufficiently close rational numbers.

Recently, Marshall gave a numerical implementation of conformal welding for finitely connected domains using the geodesic Zipper algorithm for finding the Riemann mapping between two simply connected domains and Koebe's method for computing conformal maps to circular domains [60]. Marshall then shows that if one can find the welding homeomorphisms (by using the Koebe method, for example), then one can apply the welding method to construct the maps. However, prior knowledge of the target circular domain is required for the implementation of this method.

R. Rettinger also treated the Riemann mapping theorem, its extension to multiply connected domains, and uniformization of Riemann surfaces in [88]. The main ingredient in his proof is the computational compactness of a certain class of formal power series.

4.8 Infinite Products

It has been known since the Weierstrass Factorization Theorem that some analytic functions can be expressed as infinite products of rational functions. Such factorizations may have great utility for computation as they typically converge more quickly than series. The expression of a function as an infinite product is linked to the distribution of its zeros. Here we briefly describe the computability properties of a certain kind of infinite product, the *Blaschke products*. We begin with a synopsis of the classical material.

4.8.1 Classical Background

Roughly speaking, a Blaschke product is an infinite product of conformal automorphisms of the unit disk suitably normalized. To be more precise, when $|a| < 1$, let

$$b_a(z) = \begin{cases} \frac{|a|}{a}\,\frac{z-a}{1-\overline{a}z} & a \neq 0, \\ z & a = 0. \end{cases}$$

We call the function b_a a *Blaschke factor*. If $A = \{a_n\}_{n=0}^{\infty}$ is a sequence of points in the unit disk, we let:

$$B_A = \prod_{n=0}^{\infty} b_{a_n}$$

$$\Sigma_A = \sum_{n=0}^{\infty} 1 - |a_n|.$$

The function B_A is called a *Blaschke product*, and the sum Σ_A is the corresponding *Blaschke sum*. If $\Sigma_A < \infty$, we call A a *Blaschke sequence*.

An infinite product is said to *converge* if its partial products converge to a nonzero number. It is well known (see, e.g., [92]) that if A is a Blaschke sequence, then B_A converges, except at the terms of A, to an analytic function on $\mathbb{D}$ and its partial products converge uniformly on compact subsets of the unit disk. Conversely, if B_A converges at any point in the disk besides a term of A, then A is a Blaschke sequence. Furthermore, if A is a Blaschke sequence, then the terms of A are precisely the zeros of B_A and each zero of B_A is repeated in A according to its multiplicity.

4.8.2 Computability Results

Two questions naturally arise.

1. If A is a computable Blaschke sequence, does it follow that B_A is a computable function?
2. If B_A is a computable function, does it follow that A is a computable Blaschke sequence?

It follows from the results on zeros in Subsection 4.3.3 that the answer to the second question is 'yes'. However, the answer to the first question is 'no'. For, by a result of E. Specker [96] there is a computable sequence of rational numbers $1 > r_0 > r_1 > \ldots > 0$ such that $r := \lim_n r_n$ is incomputable. If we set $a_n = \frac{r_{n+1}}{r_n}$ and $A = \{a_n\}_{n=0}^{\infty}$, then $B_A(0) = r$. It follows that A is a Blaschke sequence and that B_A is incomputable.

The question then turns to finding a sufficient and necessary parameter, and it turns out that the Blaschke sum provides the exact additional information necessary for computing B_A from A. Namely, we have the following.

Theorem 4.8.1 ([62, 64]). *Suppose A is a computable Blaschke sequence. Then, the following are equivalent.*

1. *B_A is computable.*
2. *Σ_A is computable.*
3. *B_A is computable at at least one computable point of the unit disk that is not a term of A.*

An interesting feature of Blaschke products is that their computability at a single point can yield computability throughout their domain. When we contrast this result

with Theorem 4.3.4, we see that with regards to computability Blaschke products behave much better than power series.

More material on the computability of Blaschke products can be found in [66].

4.9 Constants

4.9.1 Classical Background

Let S denote the class of functions f that are analytic and univalent in the unit disk $\mathbb{D}$ and normalized by the conditions $f(0) = f'(0) - 1 = 0$. For this class of functions, Koebe [50] proved that there is a constant c such that the set $\bigcap_{f\in S} f(\mathbb{D})$ contains a disk of radius strictly less than c. Later, Bieberbach [7] proved that $c = \frac{1}{4}$ and this result is known as Koebe's $\frac{1}{4}$-Theorem. Bloch [13] and Landau [59] proved related results for functions that are not necessary univalent (see p. 292 of [19]).

Theorem 4.9.1 (Bloch's Theorem). *If f is analytic in $\overline{\mathbb{D}}$ and $f(0) = f'(0) - 1 = 0$, then there is a disk $S \subset \mathbb{D}$ on which f is one-to-one and such that $f(S)$ contains a disk of radius $1/72$.*

Definition 4.9.2. Let $\mathscr{F}$ be the set of all functions f analytic on $\overline{\mathbb{D}}$ and $f(0) = f'(0) - 1 = 0$. For each $f \in \mathscr{F}$, let $\beta(f)$ be the supremum of all numbers b such that there is a disk $S \subset \mathbb{D}$ on which f is one-to-one and such that $f(S)$ contains a disk of radius b (hence $\beta(f) \geq 1/72$). *Bloch's constant* is

$$B = \inf\{\beta(f) : f \in \mathscr{F}\}. \tag{4.10}$$

It has been shown that

$$0.43321... = \frac{\sqrt{3}}{4} + \frac{2}{10^{14}} \leq B \leq \frac{\Gamma\left(\frac{1}{3}\right)\Gamma\left(\frac{11}{12}\right)}{(1+\sqrt{3})^{1/2}\Gamma\left(\frac{1}{4}\right)} = 0.4718617...$$

Ahlfors and Grunsky [2] conjectured that the upper bound is the exact value of Bloch's constant.

Theorem 4.9.3 (Landau's Theorem). *If f is analytic in $\overline{\mathbb{D}}$ and $f(0) = f'(0) - 1 = 0$, then $f(\mathbb{D})$ contains a disk of radius λ, where λ is an absolute constant.*

Landau proved that $\lambda \geq 1/16$.

Definition 4.9.4. Let $\mathscr{F}$ be the set of all functions f analytic on $\overline{\mathbb{D}}$ and $f(0) = f'(0) - 1 = 0$. For each $f \in \mathscr{F}$ let $\lambda(f)$ be the supremum of all numbers b such that $f(\mathbb{D})$ contains a disk of radius b (hence $\lambda(f) \geq 1/16$). *Landau's constant* is

$$L = \inf\{\lambda(f) : f \in \mathscr{F}\}. \tag{4.11}$$

It has been shown that

$$\frac{1}{2} < L \leq \frac{\Gamma\left(\frac{1}{3}\right)\Gamma\left(\frac{5}{6}\right)}{\Gamma\left(\frac{1}{6}\right)} = 0.5432589...$$

and it has been conjectured that the upper bound is the exact value of Landau's constant.

4.9.2 Computability Results

R. Rettinger proposed an algorithm for the computation of Bloch's constant [86]. The main difficulty is that the infimum over all functions in $\mathscr{F}$ cannot be computed in a finite number of steps. So he defines a subset of *test functions* $\mathscr{T}$ of $\mathscr{F}$:

$$\mathscr{T} = \{f \in \mathscr{F} : \forall z \in \mathbb{D} |f(z)| \leq c_1 \rho(z)\},$$

where $c_1 = .048 \cdot \frac{4}{\sqrt{3}}$ and $\rho(z) = \int_0^z \frac{1}{1-|t|^2} \mathrm{d}t$. He then shows that for every $\varepsilon > 0$ there is $f \in \mathscr{T}$ such that $|\beta(f) - B| < \varepsilon$ and that for the Taylor coefficients of f the inequality $|a_n| \leq \frac{c_1 \rho(\varepsilon)}{\varepsilon^n}$ holds. Utilizing the properties of the class $\mathscr{T}$, it is shown that for each precision only finitely many functions have to be considered and it is shown how to compute the $\beta(f)$'s of these functions. This leads to the following.

Theorem 4.9.5 ([86]). *Bloch's constant is computable.*

Later Rettinger developed an algorithm for the computation of Landau's constant. While similar in spirit to the algorithm for Bloch's constant, there are substantial differences due to the different nature of the problem. Again, the main result is as follows.

Theorem 4.9.6 ([89]). *Landau's constant is computable.*

4.9.3 Complexity Results

The complexity bound of the algorithms for Bloch's constant and Landau's constant is double exponential. The main obstacle to improving the time complexity of the algorithms is that the functions can explode near the boundary of the unit disk and the evaluations there can be quite time expensive.

4.10 Open Problems

Here we suggest a few directions for future inquiry.

4.10.1 The Hayman-Wu Constant

In 1981, W. Hayman and J.M.G. Wu proved the following remarkable result.

Theorem 4.10.1 (Hayman-Wu Theorem). *Suppose ϕ is a conformal map of the unit disk onto a simply connected domain D. Then, there is a constant C such that for every line L, the length of $\phi^{-1}[L \cap D]$ is no larger than C.*

The smallest such constant C is called the *Hayman-Wu constant*, which we denote by C_{HW}.

In 1992, K. Øyma gave an elegant three-page proof of the Hayman-Wu Theorem, and showed that $C_{\mathrm{HW}} \leq 4\pi$ [75]. Roughly a year later he also proved that $C_{\mathrm{HW}} \geq \pi^2$ [76]. Approximately a decade later, S. Rohde proved that $C_{\mathrm{HW}} < 4\pi$ [91]. Little else is known about this constant. Can computability theory shed some light on this situation? We are led to pose the following.

Problem 4.10.2. Is the Hayman-Wu constant computable?

4.10.2 Parameterized Complexity Riemann Mapping Theorem

As we have discussed, when it is known that a problem cannot be effectively solved, one then turns to finding additional parameters that provide the exact additional information necessary for the computation of solutions. Similarly, in complexity theory one can search for additional parameters to effect computations in polynomial time or other complexity classes. This is the field of *parameterized complexity*, which is expounded in the text of Downey and Fellows [28]. In light of the complexity results discussed in Section 4.4, we are led to the following.

Problem 4.10.3. Determine the parameterized complexity of the Riemann Mapping Theorem.

4.10.3 Computability Conformal Mapping onto Infinitely Connected Domains

In 1993, O. Schramm and Z.-X. He proved a remarkable extension of Koebe's result on circular domains.

Theorem 4.10.4 ([39]). *If a non-degenerate domain has countably many boundary components, then it is conformally equivalent to a circular domain.*

In light of the results discussed in Section 4.7, this leads to the following.

Problem 4.10.5. Suppose D is a non-degenerate domain with countably many boundary components. If D is c.e. open, and if the boundary of D is c.e. closed, does it follow that there is a computable conformal map of D onto a circular domain?

Acknowledgements We thank the anonymous reviewers for many helpful suggestions. We also thank Diego Rojas for a very thorough proofreading. The second author was supported in part by Simons Foundation grant # 317870.

References

1. Ahlfors, L.V.: Complex Analysis, third edn. *International Series in Pure and Applied Mathematics*. McGraw-Hill, New York (1978)
2. Ahlfors, L.V., Grunsky, H.: Über die Blochsche Konstante. Math. Z. **42**(1), 671–673 (1937)
3. Akhiezer, N.I.: Aerodynamical investigations. Ukrain. Akad. Nauk Trudi Fiz.-Mat. Viddilu **7**, 1–247 (1928)
4. Andreev, V., Daniel, D., McNicholl, T.H.: Computing conformal maps onto circular domains, preprint available at http://arxiv.org/abs/1307.6535
5. Andreev, V.V., McNicholl, T.H.: Estimating the error in the Koebe construction. Computational Methods and Function Theory **11**(2), 707–724 (2011)
6. Andreev, V.V., McNicholl, T.H.: Computing conformal maps of finitely connected domains onto canonical slit domains. Theory of Computing Systems **50**(2), 354–369 (2012)
7. Bieberbach, L.: Über die Koeffizienten derjenigen Potenzreihen, welche eine schlichte Abbildung des Einheitskreises vermitteln. Sitzungsber. Preuss. Akad. Wiss. Phys-Math. Kl. **1916**, 940–955 (1916)
8. Binder, I., Braverman, M.: Derandomization of euclidean random walks. In: M. Charikar, K. Jansen, O. Reingold, J.D.P. Rolim (eds.) Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, *Lecture Notes in Computer Science*, vol. 4627, pp. 353–365. Springer Berlin Heidelberg (2007)
9. Binder, I., Braverman, M., Rojas, C., Yampolsky, M.: Computability of Brolin-Lyubich measure. Comm. Math. Phys. **308**(3), 743–771 (2011)
10. Binder, I., Braverman, M., Yampolsky, M.: On the computational complexity of the Riemann mapping. Arkiv för Matematik **45**(2), 221–239 (2007)
11. Binder, I., Rojas, C., Yampolsky, M.: Computable Carathéodory theory. Adv. Math. **265**, 280–312 (2014)
12. Bishop, E., Bridges, D.S.: Constructive Analysis, *Grundlehren der Mathematischen Wissenschaften*, vol. 279. Springer, Berlin (1985)
13. Bloch, A.: Les théorèmes de M. Valiron sur les fonctions entières et la théorie de l'uniformisation. Ann. Fac. Sci. Toulouse Sci. Math. Sci. Phys. (3) **17**, 1–22 (1925)
14. Brattka, V., Weihrauch, K.: Computability on subsets of Euclidean space I: Closed and compact subsets. Theoretical Computer Science **219**, 65–93 (1999)
15. Braverman, M., Cook, S.A.: Computing over the reals: Foundations for scientific computing. Notices of the AMS **53**(3), 318–329 (2006)
16. Carathéodory, C.: Untersuchungen über die konformen Abbildungen von festen und veränderlichen Gebieten. Math. Ann. **72**(1), 107–144 (1912)
17. Carathéodory, C.: Über die gegenseitige Beziehung der Ränder bei der konformen Abbildung des Inneren einer Jordanschen Kurve auf einen Kreis. Math. Ann. **73**(2), 305–320 (1913)
18. Cheng, H.: A constructive Riemann mapping theorem. Pacific Journal Mathematics **44**, 435–454 (1973)

19. Conway, J.: Functions of One Complex Variable I, *Graduate Texts in Mathematics*, vol. 11, 2nd edn. Springer-Verlag (1978)
20. Cooper, S.B.: Computability theory. Chapman & Hall/CRC, Boca Raton, FL (2004)
21. Couch, P.J., Daniel, B.D., McNicholl, T.H.: Computing space-filling curves. Theory of Computing Systems **50**(2), 370–386 (2012)
22. Courant, R.: Dirichlet's Principle, Conformal Mapping, and Minimal Surfaces. Springer-Verlag, New York-Heidelberg (1977)
23. Courant, R., Hilbert, D.: Methods of Mathematical Physics. Vol. II. *Wiley Classics Library*. John Wiley & Sons Inc., New York (1989), reprint of the 1962 original
24. Crowdy, D.: Geometric function theory: a modern view of a classical subject. Nonlinearity **21**(10), T205–T219 (2008)
25. Daniel, D., McNicholl, T.H.: Effective versions of local connectivity properties. Theory of Computing Systems **50**(4), 621–640 (2012)
26. Davis, M.: Computability and Unsolvability. *McGraw-Hill Series in Information Processing and Computers*. McGraw-Hill, New York (1958)
27. DeLillo, T.K., Elcrat, A.R., Pfaltzgraff, J.A.: Schwarz-Christoffel mapping of multiply connected domains. J. Anal. Math. **94**, 17–47 (2004)
28. Downey, R.G., Fellows, M.R.: Fundamentals of Parameterized Complexity. Springer, London (2013)
29. Duren, P.L.: Theory of H^p Spaces, *Pure and Applied Mathematics*, vol. 38. Academic Press, New York-London (1970)
30. Gaier, D.: Untersuchung zur Durchführung der konformen Abbildung mehrfach zusammenhängender Gebiete. Arch. Rat. Mech. Anal. **3**, 149 – 178 (1959)
31. Garnett, J.B.: Bounded Analytic Functions. Academic Press, New York, London (1981)
32. Garnett, J.B., Marshall, D.E.: Harmonic Measure, *New Mathematical Monographs*, vol. 2. Cambridge University Press, Cambridge (2005)
33. Gilman, J., Kra, I., Rodriguez, R.: Complex Analysis in the Spirit of Lipman Bers, *Graduate Texts in Mathematics*, vol. 245. Springer-Verlag (2007)
34. Gordon, B.O., Julian, W., Mines, R., Richman, F.: The constructive Jordan curve theorem. Rocky Mountain Journal of Mathematics **5**, 225–236 (1975)
35. Gray, J.: On the history of the Riemann mapping theorem. Rend. Circ. Mat. Palermo (2) Suppl. **34**, 47–94 (1994)
36. Greene, R.E., Krantz, S.G.: Function Theory of One Complex Variable, *Graduate Studies in Mathematics*, vol. 40, third edn. American Mathematical Society, Providence, RI (2006)
37. Grzegorczyk, A.: On the definitions of computable real continuous functions. Fundamenta Mathematicae **44**, 61–71 (1957)
38. Harnack, A.: Grundlagen der Theorie des logarithmischen Potentiales und der eindeutigen Potentialfunktion in der Ebene. V.G. Teubner, Leipzig (1887)
39. He, Z.X., Schramm, O.: Fixed points, Koebe uniformization, and circle packings. Annals of Mathematics **137**, 369–406 (1993)
40. Henrici, P.: Applied and Computational Complex Analysis. Vol. 3. *Pure and Applied Mathematics*. John Wiley & Sons Inc., New York (1986)
41. Hertling, P.: An effective Riemann Mapping Theorem. Theoretical Computer Science **219**, 225–265 (1999)
42. Hilbert, D.: Über das Dirichletsche Prinzip. Math. Ann. **59**(1-2), 161–186 (1904)
43. Hilbert, D.: Über das Dirichletsche Prinzip. J. Reine Angew. Math. **129**, 63–67 (1905)
44. Hocking, J.G., Young, G.S.: Topology, second edn. Dover Publications Inc., New York (1988)
45. Kakutani, S.: Two-dimensional Brownian motion and harmonic functions. Proc. Imp. Acad. Tokyo **20**, 706–714 (1944)
46. Kalantari, I., Welch, L.: Point-free topological spaces, functions and recursive points; filter foundation for recursive analysis. I. Annals of Pure and Applied Logic **93**(1–3), 125–151 (1998)
47. Kalantari, I., Welch, L.: Recursive and nonextendible functions over the reals; filter foundation for recursive analysis, II. Annals of Pure and Applied Logic **98**(1–3), 87–110 (1999)

48. Klein, F.: Neue Beiträge zur Riemann'schen Functionentheorie. Math. Ann. **21**(2), 141–218 (1883)
49. Koebe, P.: Über die Uniformisierung beliebiger analytischer Kurven. Nachr. Ges. Wiss. Göttingen, Math.-Phys. Kl. **1907**, 191–210 (1907)
50. Koebe, P.: Über die Uniformisierung beliebiger analytischer Kurven (Zweite Mitteilung). Nachr. Ges. Wiss. Göttingen, Math.-Phys. Kl. **1907**, 633–669 (1907)
51. Koebe, P.: Über die Uniformisierung reeller algebraischer Kurven. Nachr. Ges. Wiss. Göttingen, Math.-Phys. Kl. **1907**, 177–190 (1907)
52. Koebe, P.: Über eine neue Methode der Konformen Abbildung und Uniformisierung. Nachr. Ges. Wiss. Göttingen, Math.-Phys. Kl. **1912**, 844–848 (1912)
53. Koebe, P.: Ränderzuordnung bei konformer Abbildung. Nachr. Ges. Wiss. Göttingen, Math.-Phys. Kl. **1913**, 286–288 (1913)
54. Koebe, P.: Abhandlungen zur Theorie der Konformen Abbildung, IV. Acta Math. **41**, 305–344 (1918)
55. Kühnau, R.: Numerische Realisierung konformer Abbildungen durch "Interpolation". Z. Angew. Math. Mech. **63**(12), 631–637 (1983)
56. Lacombe, D.: Extension de la notion de fonction récursive aux fonctions d'une ou plusieurs variables réelles I. Comptes Rendus Académie des Sciences Paris **240**, 2478–2480 (1955)
57. Lacombe, D.: Extension de la notion de fonction récursive aux fonctions d'une ou plusieurs variables réelles II. Comptes Rendus Académie des Sciences Paris **241**, 13–14 (1955)
58. Lacombe, D.: Extension de la notion de fonction récursive aux fonctions d'une ou plusieurs variables réelles III. Comptes Rendus Académie des Sciences Paris **241**, 151–153 (1955)
59. Landau, E.: Der Picard-Schottkysche Satz und die Blochsche Konstante. Sitzungsber. Preuss. Akad. Wiss. Phys-Math. Kl. pp. 467–474 (1926)
60. Marshall, D.E.: Conformal welding for finitely connected regions. Comput. Methods Funct. Theory **11**(2), 655–669 (2011)
61. Marshall, D.E., Rohde, S.: Convergence of a variant of the Zipper algorithm for conformal mapping. SIAM J. Numer. Anal. **45**(6), 2577–2609 (2007)
62. Matheson, A., McNicholl, T.H.: Computable analysis and Blaschke products. Proceedings of the American Mathematical Society **136**(1), 321–332 (2008)
63. McNicholl, T.: Computing boundary extensions of conformal maps part 2 (2014), preprint available at https://arxiv.org/abs/1304.1915
64. McNicholl, T.H.: Uniformly computable aspects of inner functions: estimation and factorization. Mathematical Logic Quarterly **54**(5), 508–518 (2008)
65. McNicholl, T.H.: An effective Carathéodory theorem. Theory of Computing Systems **50**(4), 579–588 (2012)
66. McNicholl, T.H.: On the computable theory of bounded analytic functions. In: Blaschke Products and Their Applications, *Fields Inst. Commun.*, vol. 65, pp. 223–248. Springer, New York (2013)
67. McNicholl, T.H.: An operator-theoretic existence proof of solutions to planar Dirichlét problems. Complex Anal. Oper. Theory **7**(4), 1311–1319 (2013)
68. McNicholl, T.H.: Computing boundary extensions of conformal maps. LMS Journal of Computation and Mathematics **17**(1), 360–378 (2014)
69. McNicholl, T.H.: Carathéodory's theorem and moduli of local connectivity. Complex Variables and Elliptic Equations **61**(1), 76–85 (2016)
70. Miller, J.S.: Effectiveness for embedded spheres and balls. In: V. Brattka, M. Schröder, K. Weihrauch (eds.) CCA 2002 Computability and Complexity in Analysis, *Electronic Notes in Theoretical Computer Science*, vol. 66. Elsevier, Amsterdam (2002), 5th International Workshop, CCA 2002, Málaga, Spain, July 12–13, 2002
71. Myhill, J.: A recursive function defined on a compact interval and having a continuous derivative that is not recursive. Michigan Math. J. **18**, 97–98 (1971)
72. Nehari, Z.: Conformal Mapping. McGraw-Hill, New York, Toronto, London (1952)
73. Neumann, C.: Zur Theorie des Potentiales. Math. Ann. **2**(3), 514 (1870)
74. Osgood, W.F.: On the existence of the Green's function for the most general simply connected plane region. Trans. Amer. Math. Soc. **1**(3), 310–314 (1900)

75. Øyma, K.: Harmonic measure and conformal length. Proc. Amer. Math. Soc. **115**(3), 687–689 (1992)
76. Øyma, K.: The Hayman-Wu constant. Proc. Amer. Math. Soc. **119**(1), 337–338 (1993)
77. Poincare, H.: Mémoire sur les fonctions fuchsiennes. Acta Math. **1**(1), 193–294 (1882)
78. Poincaré, H.: Sur les groupes des équations linéaires. Acta Math. **4**(1), 201–312 (1884)
79. Poincaré, H.: Sur l'uniformisation des fonctions analytiques. Acta Math. **31**(1), 1–63 (1908)
80. Pommerenke, C.: Boundary Behaviour of Conformal Maps, *Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*, vol. 299. Springer-Verlag, Berlin (1992)
81. Pour-El, M.B., Caldwell, J.: On a simple definition of computable functions of a real variable - with applications to functions of a complex variable. Zeitschrift für Mathematische Logik und Grundlagen der Mathematik **21**, 1–19 (1975)
82. Pour-El, M.B., Richards, J.I.: Computability in Analysis and Physics. *Perspectives in Mathematical Logic*. Springer, Berlin (1989)
83. Rado, T.: Über die Fundamentalabbildung schlichter Gebiete. Acta Szeged **I**, 240–251 (1922–1923)
84. Ransford, T., Rostand, J.: Computation of capacity. Mathematics of Computation **76**(259), 1499–1520 (2007)
85. Remmert, R.: Classical Topics in Complex Function Theory, *Graduate Texts in Mathematics*, vol. 172. Springer-Verlag, New York (1998)
86. Rettinger, R.: Bloch's constant is computable. Journal of Universal Computer Science **14**(6), 896–907 (2008)
87. Rettinger, R.: Computability and complexity aspects of univariate complex analysis. Habilitation thesis. FernUniversität Hagen (2008)
88. Rettinger, R.: Compactness and the effectivity of uniformization. In: S.B. Cooper, A. Dawar, B. Löwe (eds.) How the World Computes, *Lecture Notes in Computer Science*, vol. 7318, pp. 626–625. Springer, Heidelberg (2012)
89. Rettinger, R.: On computable approximations of Landau's constant. Log. Methods Comput. Sci. **8**(4), 4:15, 11 (2012)
90. Rogers, H., Jr.: Theory of Recursive Functions and Effective Computability, second edn. MIT Press, Cambridge (1987)
91. Rohde, S.: On the theorem of Hayman and Wu. Proc. Amer. Math. Soc. **130**(2), 387–394 (2002)
92. Rudin, W.: Real and Complex Analysis, 3rd edn. McGraw-Hill, New York (1987)
93. Schiffer, M.: Some recent developments in the theory of conformal mapping. Appendix to: R. Courant. Dirichlet Principle, Conformal Mapping and Minimal Surfaces, Interscience Publishers (1950)
94. Schwarz, H.A.: Ueber einige Abbildungsaufgaben. J. Reine Angew. Math. **70**, 105–120 (1869)
95. Soare, R.I.: Recursively Enumerable Sets and Degrees. *Perspectives in Mathematical Logic*. Springer, Berlin (1987)
96. Specker, E.: Nicht konstruktiv beweisbare Sätze der Analysis. The Journal of Symbolic Logic **14**(3), 145–158 (1949)
97. Specker, E.: Der Satz vom Maximum in der rekursiven Analysis. In: A. Heyting (ed.) Constructivity in Mathematics, *Studies in Logic and the Foundations of Mathematics*, pp. 254–265. North-Holland, Amsterdam (1959), proc. Colloq., Amsterdam, Aug. 26–31, 1957
98. Turing, A.M.: On computable numbers, with an application to the "Entscheidungsproblem". Proceedings of the London Mathematical Society **42**(2), 230–265 (1937)
99. Walsh, J.L.: History of the Riemann mapping theorem. Amer. Math. Monthly **80**, 270–276 (1973)
100. Weierstrass, K.: Über das sogenannte Dirichletsche Princip. In: Mathematische Werke. II. Abhandlungen 2, pp. 49–54. Georg Olms Verlagsbuchhandlung, Hildesheim; Johnson Reprint Corp., New York (1967)
101. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)

102. Zhou, Q.: Computable real-valued functions on recursive open and closed subsets of Euclidean space. Mathematical Logic Quarterly **42**, 379–409 (1996)
103. Ziegler, M.: Effectively open real functions. Journal of Complexity **22**(6), 827–849 (2006)

Part II
Complexity, Dynamics, and Randomness

Chapter 5
Computable Geometric Complex Analysis and Complex Dynamics

Cristóbal Rojas, Michael Yampolsky

Abstract We discuss computability and computational complexity of conformal mappings and their boundary extensions. As applications, we review the state of the art regarding computability and complexity of Julia sets, their invariant measures, and external ray impressions.

5.1 Introduction

The purpose of this chapter is to survey the exciting recent applications of Computable Analysis to Geometric Complex Analysis and Complex Dynamics. The first effort at a systematic development of Computable Analysis was made by Banach and Mazur [1] in 1937, only one year after the birth of Turing Machines and Post Machines ([60, 54]). Interrupted by war, this work was further developed in the book by Mazur [44], and followed in the mid-1950s by the works of Grzegorczyk [30], Lacombe [38], and others. A parallel school of Constructive Analysis was founded by A. A. Markov in Russia in the late 1940s. A modern treatment of the field can be found in [35] and [61].

In the last two decades, there has been much activity both by theorists and by computational practitioners in applying natural notions of computational hardness to such complex-analytic objects as the Riemann Mapping, Carathéodory Extension, the Mandelbrot set, and Julia sets. The goal of this survey is to give a brief summary of the beautiful interplay between Computability, Analysis, and Geometry which has emerged from these works.

Cristóbal Rojas
Departamento de Matemáticas, Universidad Andrés Bello, Santiago, Chile
e-mail: crojas@mat-unab.cl

Michael Yampolsky
Department of Mathematics, University of Toronto, Canada
e-mail: yampol@math.toronto.edu

V. Brattka, P. Hertling (eds.), *Handbook of Computability and Complexity in Analysis*,
Theory and Applications of Computability, https://doi.org/10.1007/978-3-030-59234-9_5

5.2 Required Computability Notions

In this section we gather all the computability notions involved in the statements presented in the survey; the reader is referred to [61] for the details. We note that in what follows, $\mathbb{N}$ denotes the set of positive integers: $\mathbb{N} = \{1, 2, \ldots\}$.

5.2.1 Computable Metric Spaces

Definition 5.2.1. A *computable metric space* is a triple $(X, d, \mathscr{S})$ where:

1. (X, d) is a separable metric space,
2. $\mathscr{S} = \{s_i : i \in \mathbb{N}\}$ is a dense sequence of points in X,
3. $d(s_i, s_j)$ are computable real numbers, uniformly in (i, j).

The points in $\mathscr{S}$ are called *ideal*.

Example 5.2.2. A basic example is to take the space $X = \mathbb{R}^n$ with the usual notion of Euclidean distance $d(\cdot,\cdot)$, and to let the set $\mathscr{S}$ consist of points $\bar{x} = (x_1, \ldots, x_n)$ with rational coordinates. In what follows, we will implicitly make these choices of $\mathscr{S}$ and $d(\cdot,\cdot)$ when discussing computability in $\mathbb{R}^n$.

Definition 5.2.3. A point x is *computable* if there is a computable function $f : \mathbb{N} \to \mathscr{S}$ such that

$$d(f(n), x) < 2^{-n} \text{ for all } n.$$

If $x \in X$ and $r > 0$, the metric ball $B(x, r)$ is defined as

$$B(x, r) = \{y \in X : d(x, y) < r\}.$$

Since the set $\mathscr{B} := \{B(s, q) : s \in \mathscr{S}, q \in \mathbb{Q}, q > 0\}$ of *ideal balls* is countable, we can fix an enumeration $\mathscr{B} = \{B_i : i \in \mathbb{N}\}$, which we assume to be effective with respect to the enumerations of $\mathscr{S}$ and $\mathbb{Q}$.

Definition 5.2.4. An open set U is called *lower-computable* if there is a computable function $f : \mathbb{N} \to \mathbb{N}$ such that

$$U = \bigcup_{n \in \mathbb{N}} B_{f(n)}.$$

It is not difficult to see that finite intersections or infinite unions of (uniformly) lower-computable open sets are again lower-computable.

Having defined lower-computable open sets, we naturally proceed to the following definitions for closed sets.

Definition 5.2.5. A closed set K is called *upper-computable* if its complement is a lower-computable open set, and *lower-computable* if the relation $K \cap B_i \neq \emptyset$ is semi-decidable, uniformly in i.

In other words, a closed set K is lower-computable if there exists an algorithm $\mathscr{A}$ which enumerates all ideal balls which have non-empty intersection with K. To see that this definition is a natural extension of lower-computability of open sets, we note the following.

Example 5.2.6. The closure $\overline{U}$ of any open lower-computable set U is lower-computable since $B_i \cap \overline{U} \neq \emptyset$ if and only if there exists $s \in B_i \cap U$, which is uniformly semi-decidable for such U.

The following is a useful characterization of lower-computable sets.

Proposition 5.2.7. *A closed set K is lower-computable if and only if there exists a sequence of uniformly computable points $x_i \in K$ which is dense in K.*

Proof. Observe that, given some ideal ball $B = B(s,q)$ intersecting K, the relations $\overline{B_i} \subset B$, $q_i < 2^{-k}$ and $B_i \cap K \neq \emptyset$ are all semi-decidable, so we can find an exponentially decreasing sequence of ideal balls (B_k) intersecting K. Hence $\{x\} = \cap_k B_k$ is a computable point lying in $B \cap K$.

The other direction is obvious.

Definition 5.2.8. A closed set is *computable* if it is lower and upper computable.

For compact sets, there is an alternative way to define computability. We will write $K \Subset X$ to mean that K is a compact subset of X. Recall that the *Hausdorff distance* between two nonempty compact sets K_1, K_2 is

$$\mathrm{dist}_H(K_1,K_2) = \inf_{\varepsilon}\{K_1 \subset U_\varepsilon(K_2) \text{ and } K_2 \subset U_\varepsilon(K_1)\},$$

where $U_\varepsilon(K) = \bigcup_{z\in K} B(z,\varepsilon)$ stands for the ε-neighborhood of a set. The set of all compact subsets of X equipped with the Hausdorff distance is a metric space which we will denote by $\mathscr{K}(X)$. If X is a computable metric space, then $\mathscr{K}(X)$ inherits this property; the ideal points in $\mathscr{K}(X)$ can be taken to be, for instance, finite unions of closed ideal balls in X. We then have the following.

Proposition 5.2.9. *A compact set $K \Subset X$ is computable if and only if it is a computable point in $\mathscr{K}(X)$.*

Proposition 5.2.10. *Equivalently, a compact set K is computable if there exists an algorithm $\mathscr{A}$ with a single natural input n which outputs a finite collection of closed ideal balls $\overline{B_1},\ldots,\overline{B_{i_n}}$ such that*

$$\mathrm{dist}_H(\bigcup_{j=1}^{i_n} \overline{B_j}, K) < 2^{-n}.$$

We introduce the following computable version of compactness.

Definition 5.2.11. A set $K \subseteq X$ is called *computably compact* if it is compact and there exists an algorithm $\mathscr{A}$ which on input $(i_1,\ldots,i_p)$ halts if and only if $(B_{i_1},\ldots,B_{i_p})$ is a covering of K.

In other words, a compact set K is computably compact if we can semi-decide the inclusion $K \subset U$, uniformly from a description of the open set U as a lower-computable open set. It is not hard to see that when the space X is computably compact, then the collection of subsets of X which are computably compact coincides with the collection of upper-computable closed sets. As an example, it is easy to see that a singleton $\{x\}$ is a computably compact set if and only if x is a computable point.

We end this section with the definition of a computable function between computable metric spaces. Let X' be another computable metric space with ideal points $\mathscr{S}'$ and ideal balls $\mathscr{B}' = \{B'_i\}$.

Definition 5.2.12. A function $f : X \to X'$ is *computable* if the sets $f^{-1}(B'_i)$ are lower-computable open, uniformly in i.

An immediate corollary of the definition is as follows.

Proposition 5.2.13. *Every computable function is continuous.*

The above definition of a computable function is concise, yet not very transparent. To give its $\varepsilon - \delta$ version, we need another concept. We say that a function $\phi : \mathbb{N} \to \mathbb{N}$ is an *oracle* for $x \in X$ if

$$d(s_{\phi(m)}, x) < 2^{-m}.$$

An algorithm may *query* an oracle by reading the values of the function ϕ for an arbitrary $n \in \mathbb{N}$. We have the following (see, e.g., [61]).

Proposition 5.2.14. *A function $f : X \to X'$ is computable if and only if there exists an algorithm $\mathscr{A}$ with an oracle for $x \in X$ and an input $n \in \mathbb{N}$ which outputs $s'_n \in \mathscr{S}'$ such that $d(s'_n, f(x)) < 2^{-n}$.*

In other words, given an arbitrarily good approximation of the input of f it is possible to constructively approximate the value of f with any desired precision.

5.2.2 Computability of Probability Measures

Let $\mathscr{M}(X)$ denote the set of Borel probability measures over a metric space X, which we will assume to be endowed with a computable structure for which it is computably compact. We recall the notion of weak convergence of measures.

Definition 5.2.15. A sequence of measures $\mu_n \in \mathscr{M}(X)$ is said to be *weakly convergent* to $\mu \in \mathscr{M}(X)$ if $\int f \mathrm{d}\mu_n \to \int f \mathrm{d}\mu$ for each $f \in C_0(X)$.

It is well known that when X is a compact separable and complete metric space, then so is $\mathscr{M}(X)$. In this case, weak convergence on $\mathscr{M}(X)$ is compatible with the notion of *Wasserstein-Kantorovich distance*, defined by:

$$W_1(\mu,\nu) = \sup_{f\in 1\text{-Lip}(X)} \left| \int f \mathrm{d}\mu - \int f \mathrm{d}\nu \right|$$

where 1-Lip(X) is the space of Lipschitz functions on X having Lipschitz constant less than one.

The following result (see [33]) says that the computable metric structure on X is inherited by $\mathscr{M}(X)$.

Proposition 5.2.16. *Let $\mathscr{D}$ be the set of finite convex rational combinations of Dirac measures supported on ideal points of the computable metric space X. Then the triple $(\mathscr{M}(X), W_1, \mathscr{D})$ is a computable metric space.*

Definition 5.2.17. A *computable measure* is a computable point in $(\mathscr{M}(X), W_1, \mathscr{D})$. That is, it is a measure which can be algorithmically approximated in the weak sense by *discrete* measures with any given precision.

The following proposition (see [33]) brings the previous definition to a more familiar setting.

Proposition 5.2.18. *A probability measure μ on X is computable if and only if one can uniformly compute integrals of computable functions.*

We also note the following (see, e.g., [29]).

Proposition 5.2.19. *The support of a computable measure is a lower-computable set.*

As examples of computable measures we mention Lebesgue measure in $[0,1]^n$, or any smooth measure in $[0,1]^n$ with a computable density function.

5.2.3 Time Complexity of a Problem

For an algorithm $\mathscr{A}$ with input w the *running time* is the number of steps $\mathscr{A}$ makes before terminating with an output. The *size* of an input w is the number of dyadic bits required to specify w. Thus for $w \in \mathbb{N}$, the size of w is the smallest integer $l(w) \geq \log_2 w$. The *running time of* $\mathscr{A}$ is the function

$$T_{\mathscr{A}} : \mathbb{N} \to \mathbb{N}$$

such that

$$T_{\mathscr{A}}(n) = \max\{\text{the running time of } \mathscr{A}(w) \text{ for inputs } w \text{ of size } n\}.$$

In other words, $T_{\mathscr{A}}(n)$ is the worst-case running time for inputs of size n. For a computable function $f : \mathbb{N} \to \mathbb{N}$ the time complexity of f is said to have an upper

bound $T(n)$ if there exists an algorithm $\mathscr{A}$ with running time bounded by $T(n)$ that computes f. We say that the time complexity of f has a lower bound $T(n)$ if for every algorithm $\mathscr{A}$ which computes f, there is a subsequence n_k such that the running time

$$T_{\mathscr{A}}(n_k) > T(n_k).$$

5.2.4 Computational Complexity of Two-Dimensional Images

Intuitively, we say the time complexity of a set $S \subset \mathbb{R}^2$ is $t(n)$ if it takes time $t(n)$ to decide whether to draw a pixel of size 2^{-n} in the picture of S. Mathematically, the definition is as follows.

Definition 5.2.20. A set T is said to be a 2^{-n}-picture of a bounded set $S \subset \mathbb{R}^2$ if:

(i) $S \subset T$, and
(ii) $T \subset B(S, 2^{-n}) = \{x \in \mathbb{R}^2 \ : \ |x-s| < 2^{-n} \text{ for some } s \in S\}$.

Definition 5.2.20 means that T is a 2^{-n}-approximation of S with respect to the Hausdorff metric.

Suppose we are trying to generate a picture of a set S using a union of round pixels of radius 2^{-n} with centers at all the points of the form $\left(\frac{i}{2^n}, \frac{j}{2^n}\right)$, with i and j integers. In order to draw the picture, we have to decide for each pair (i, j) whether to draw the pixel centered at $\left(\frac{i}{2^n}, \frac{j}{2^n}\right)$ or not. We want to draw the pixel if it intersects S and to omit it if some neighborhood of the pixel does not intersect S. Formally, we want to compute a function

$$f_S(n, i/2^n, j/2^n) = \begin{cases} 1, & B((i/2^n, j/2^n), 2^{-n}) \cap S \neq \emptyset \\ 0, & B((i/2^n, j/2^n), 2 \cdot 2^{-n}) \cap S = \emptyset \\ 0 \text{ or } 1, & \text{in all other cases} \end{cases} \tag{5.1}$$

The time complexity of S is defined as follows.

Definition 5.2.21. A bounded set S is said to be computable in time $t(n)$ if there is a function $f(n, \cdot)$ satisfying (5.1) which runs in time $t(n)$. We say that S is poly-time computable if there is a polynomial p such that S is computable in time $p(n)$.

Computability of sets in bounded space is defined in a similar manner. There, the amount of *memory* the machine is allowed to use is restricted.

To see why this is the "right" definition, suppose we are trying to draw a set S on a computer screen which has a 1000×1000 pixel resolution. A 2^{-n}-zoomed in picture of S has $O(2^{2n})$ pixels of size 2^{-n}, and thus would take time $O(t(n) \cdot 2^{2n})$ to compute. This quantity is exponential in n, even if $t(n)$ is bounded by a polynomial. But we are drawing S on a finite-resolution display, and we will only need to draw $1000 \cdot 1000 = 10^6$ pixels. Hence the running time would be $O(10^6 \cdot t(n)) = O(t(n))$. This running time is polynomial in n if and only if $t(n)$ is polynomial. Hence $t(n)$ reflects the 'true' cost of zooming in.

5.3 Computability and Complexity of Conformal Mappings

Let $\mathbb{D} = \{|z| < 1\} \subset \mathbb{C}$. The celebrated Riemann Mapping Theorem asserts the following.

Riemann Mapping Theorem. *Suppose $W \subsetneq \mathbb{C}$ is a simply connected domain in the complex plane, and let w be an arbitrary point in W. Then there exists a unique conformal mapping*

$$f : \mathbb{D} \to W, \text{ such that } f(0) = w \text{ and } f'(0) > 0.$$

The inverse mapping,

$$\varphi \equiv f^{-1} : W \to \mathbb{D}$$

is called the Riemann mapping of the domain W with base point w. The first complete proof of the Riemann Mapping Theorem was given by Osgood [50] in 1900. The first constructive proof of the Riemann Uniformization Theorem is due to Koebe [36], and dates to the early 1900s. Formal proofs of the constructive nature of the Theorem which follow Koebe's argument under various computability conditions on the boundary of the domain are numerous in the literature (see, e.g., [19, 9, 64, 31]).

The following theorem, due to Hertling [31], characterizes the information required from the domain W in order to compute the Riemann map.

Theorem 5.3.1. *Let $W \subsetneq \mathbb{C}$ be a simply connected domain. Then, the following are equivalent:*

(i) W is a lower-computable open set, ∂W is a lower-computable closed set, and $w \in W$ is a computable point.
(ii) The conformal bijection

$$f : \mathbb{D} \to W,\ f(0) = w,\ f'(0) > 0; \text{ and its inverse } \varphi \equiv f^{-1} : W \to \mathbb{D},$$

are both computable.

We now move to discussing the computational complexity of computing the Riemann map. For this analysis we assume the domain W is computable.

For $w \in W$ as above, the quantity $1/|\varphi'(w)|$ is called the *conformal radius* of W at w. In [6] it was shown that even if the domain we are uniformizing is very simple computationally, the complexity of the uniformization can be quite high. In fact, it might already be difficult to compute the conformal radius of the domain.

Theorem 5.3.2. *Suppose there is an algorithm A that given a simply connected domain W with a linear-time computable boundary and an inner radius $> \frac{1}{2}$ and a number n computes the first $20n$ digits of the conformal radius $r(W,0)$, then we can use one call to A to solve any instance of $\#SAT(n)$ with a linear time overhead.*

In other words, $\#\mathbf{P}$ is poly-time reducible to computing the conformal radius of a set.

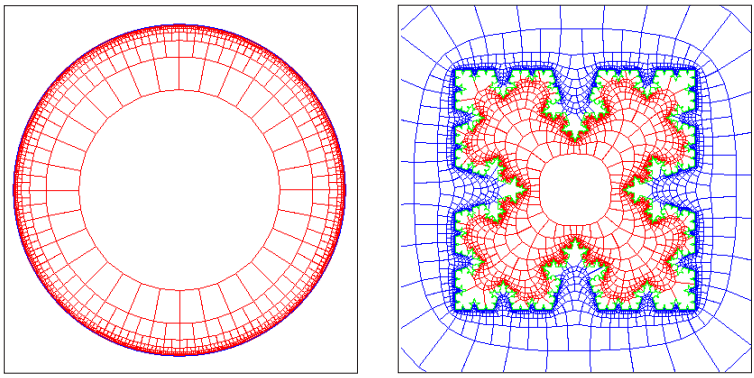


Fig. 5.1 An example of the work of "Zipper". On the left: a standard Carleson grid on the disk. On the right: the image of the Carleson grid by the conformal map to the interior of the inverted snowflake and the image by the conformal map to the exterior of the snowflake. Both images taken from D. Marshall's web page [42].

Rettinger in [56] showed that this complexity bound is sharp.

Theorem 5.3.3. *Assume that the boundary ∂W is given by an oracle (that is, we can query an oracle for a function of the form* (5.1)*). Let us also have an oracle access to w, and to another point $z \in W$. Then the problem of computing $\varphi(z)$ with precision n is in* $\#\mathbf{P}(n)$.

This result of [56] improved that of [6], where the same statement was obtained with complexity bound **PSPACE**. The statement of Theorem 5.3.3 (as well as its predecessor in [6]) is not based on an explicit algorithm. Rather, the existence of A is derived from derandomization results for random walks established in [2].

Computation of the mapping φ is important for applications, and numerous algorithms have been implemented in practice; however, none of them reaches the theoretical efficiency limit of $\#\mathbf{P}$. The most computationally efficient algorithm used nowadays to calculate the conformal map is the "Zipper", invented independently by Marshall [42] and Kühnau [37]; for examples see Figure 5.1. The effectiveness of this algorithm was studied by Marshall and Rohde in [43]. In the setting of Theorem 5.3.3, it belongs to the complexity class $\mathbf{EXP}(n)$. It is reasonable to expect then, that an algorithm can be found in the class $\#\mathbf{P}$ which is more practically efficient than "Zipper".

5.4 Computable Carathéodory Theory

The theory of Carathéodory (see, e.g., [49, 53]) deals with the question of extending the map f to the unit circle. It is most widely known in the case when ∂W is a locally

connected set. We remind the reader that a Hausdorff topological space X is called locally connected if for every point $x \in X$ and every open set $V \ni x$ there exists a connected set $U \subset V$ such that x lies in the interior of U. Thus, every point $x \in X$ has a basis of connected, but not necessarily open, neighborhoods. This condition is easily shown to be equivalent to the (seemingly stronger) requirement that every point $x \in X$ has a basis of open connected neighborhoods. In its simplest form, Carathéodory Theorem says the following.

Carathéodory Theorem for locally connected domains. *A conformal mapping $f : \mathbb{D} \to W$ continuously extends to the unit circle if and only if ∂W is locally connected.*

A natural question from the point of view of Computability Theory is then the following:

What information do we need about the boundary of the domain W in order to compute the Carathéodory extension $f : S^1 \to \partial W$?

Below we discuss a constructive Carathéodory theory, which, in particular, answers this question.

5.4.1 Carathéodory Extension Theorem

We give a very brief account of the principal elements of the theory here, for details see, e.g., [49, 53]. In what follows, we fix a bounded simply connected domain $W \subset \mathbb{C}$, and a point $w \in W$; we will refer to such a pair as a *pointed domain*, and use notation (W, w) (to adapt what follows to unbounded domains, one simply needs to replace the Euclidean metric on $\mathbb{C}$ with the spherical metric on $\hat{\mathbb{C}}$). A *crosscut* $\gamma \subset W$ is a homeomorphic image of the open interval $(0,1)$ such that the closure $\overline{\gamma}$ is homeomorphic to the closed interval $[0,1]$ and the two endpoints of $\overline{\gamma}$ lie in ∂W. It is not difficult to see that a crosscut divides W into two connected components. Let γ be a crosscut such that $w \notin \gamma$. The component of $W \setminus \gamma$ which does not contain w is called *the crosscut neighborhood* of γ in (W, w). We will denote it N_γ.

A *fundamental chain* in (W, w) is a nested infinite sequence

$$N_{\gamma_1} \supset N_{\gamma_2} \supset N_{\gamma_3} \supset \cdots$$

of crosscut neighborhoods such that the closures of the crosscuts γ_j are disjoint, and such that

$$\operatorname{diam} \gamma_j \longrightarrow 0.$$

Two fundamental chains $(N_{\gamma_j})_{j=1}^{\infty}$ and $(N_{\tau_j})_{j=1}^{\infty}$ are *equivalent* if every N_{γ_j} contains some N_{τ_i} and conversely, every N_{τ_i} contains some N_{γ_j}. Note that any two fundamental chains $(N_{\gamma_j})_{j=1}^{\infty}$ and $(N_{\tau_j})_{j=1}^{\infty}$ are either equivalent or eventually disjoint, i.e., $N_{\gamma_j} \cap N_{\tau_i} = \emptyset$ for i and j sufficiently large.

The key concept of Carathéodory theory is a *prime end*, which is an equivalence class of fundamental chains. The *impression* $\mathscr{I}(p)$ of a prime end p is a compact connected subset of ∂W defined as follows: let $(N_{\gamma_j})_{j=1}^{\infty}$ be any fundamental chain in the equivalence class p, then

$$\mathscr{I}(p) = \cap \overline{N_{\gamma_j}}.$$

We say that the impression of a prime end is *trivial* if it consists of a single point. It is easy to see (cf. [49]) the following.

Proposition 5.4.1. *If the boundary ∂W is locally connected then the impression of every prime end is trivial.*

We define the *Carathéodory compactification* $\widehat{W}$ to be the disjoint union of W and the set of prime ends of W with the following topology. For any crosscut neighborhood N let $\widetilde{N} \subset \widehat{W}$ be the neighborhood N itself, and the collection of all prime ends which can be represented by fundamental chains starting with N. These neighborhoods, together with the open subsets of W, form the basis for the topology of $\widehat{W}$. The above definition originated in [45].

Carathéodory Theorem. *Every conformal isomorphism $\phi : W \to \mathbb{D}$ extends uniquely to a homeomorphism*

$$\hat{\phi} : \widehat{W} \to \overline{\mathbb{D}}.$$

The Carathéodory Theorem for locally connected domains is a synthesis of the above statement and Proposition 5.4.1.

Let us note the following (see [49], p. 184).

Lemma 5.4.2. *If f is a continuous map from a compact locally connected space X onto a Hausdorff space Y, then Y is also locally connected.*

Theorem 5.4.3. *In the case when W is Jordan, the identity map $W \to W$ extends to a homeomorphism between the Carathéodory compactification $\widehat{W}$ and $\overline{W}$.*

In the Jordan case, we will use the notation $\overline{\phi}$ for the extension of a conformal map to the closure of W. Of course,

$$\overline{\phi} = (\overline{f})^{-1}.$$

The Carathéodory compactification of (W, w) can be seen as its metric completion for the following metric. Let z_1, z_2 be two points in W distinct from w. We will define the *crosscut distance* $\mathrm{dist}_C^W(z_1, z_2)$ between z_1 and z_2 as the infimum of the diameters of curves γ in W for which one of the following properties holds:

- γ is a crosscut such that z_1 and z_2 are contained in the crosscut neighborhood N_γ;
- γ is a simple closed curve such that z_1 and z_2 are contained in the bounded component of the complement $\mathbb{C} \setminus \gamma$ and γ separates z_1, z_2 from w.

The following is easy to verify.

Theorem 5.4.4. *The crosscut distance is a metric on $W \setminus \{w\}$ which is locally equal to the Euclidean one. The completion of W equipped with* dist_C *is homeomorphic to* $\widehat{W}$.

5.4.2 Computational Representation of Prime Ends

Definition 5.4.5. We say that a curve $g : (0,1) \to \mathbb{C}$ is a *rational polygonal curve* if:

- the image of g is a simple curve;
- g is piecewise-linear with rational coefficients.

The following is elementary.

Proposition 5.4.6. *Let N_{γ_j} be a fundamental chain in a pointed simply connected domain (W,w). Then there exists an equivalent fundamental chain N_{τ_j} such that the following holds. For every j there exists a rational polygonal curve $t_j : (0,1) \to \mathbb{C}$ with*

$$t_j(0.5) \in N_{\gamma_j} \setminus \overline{N_{\gamma_{j+1}}},$$

and such that $\tau_j \subset t_j([0.01, 0.99])$. Furthermore, t_j can be chosen so that

$$\operatorname{diam} t_j(0,1) \underset{j\to\infty}{\longrightarrow} 0.$$

We call the sequence of polygonal curves t_j as described in the above proposition a *representation* of the prime end p specified by N_{γ_j}. Since only a finite amount of information suffices to describe each rational polygonal curve t_j, the sequence t_j can be specified by an oracle. Namely, there exists an algorithm $\mathscr{A}$ such that for every representation (t_j) of a prime end there exists a function $\phi : \mathbb{N} \to \mathbb{N}$ such that given access to the values of $\phi(i)$, $i = 1, \ldots, n$, the algorithm $\mathscr{A}$ outputs the coefficients of the rational polygonal curves t_j, $j = 1, \ldots, m_n$ with $m_n \underset{n\to\infty}{\longrightarrow} \infty$. We will refer to such a ϕ simply as *an oracle for p*.

5.4.3 Structure of a Computable Metric Space on the Carathéodory Compactification

Let $K \Subset \mathbb{C}$. We say that ϕ is *an oracle for K* if ϕ is a function from the natural numbers to sets of finite sequences of triples (x_j, y_j, r_j) of rational numbers with the following property. Let

$$\phi(n) = \{(x_j, y_j, r_j)\}_{j=1}^{k_n},$$

and let B_j be the ball of radius r_j about the point $x_j + iy_j$. Then

$$\mathrm{dist}_H(\bigcup_{j=1}^{k_n} \overline{B_j}, K) < 2^{-n}.$$

Let (W,w) be a simply connected pointed domain. Then the following conditional computability result holds (see [7]).

Theorem 5.4.7. *The following is true in the presence of oracles for w and for ∂W. The Carathéodory completion $\widehat{W}$ equipped with the crosscut distance and with rational points in W as ideal points is a computable metric space. Moreover, this space is computably compact.*

5.4.4 Moduli of Locally Connected Domains

The following definition is standard.

Definition 5.4.8 (Modulus of local connectivity). Let $X \subset \mathbb{R}^2$ be a connected set. Any strictly increasing function $m : (0,a) \to \mathbb{R}$ is called a *modulus of local connectivity* of X if

- for all $x,y \in X$ such that $\mathrm{dist}(x,y) < r < a$ there exists a connected subset $L \subset X$ containing both x and y with the property $\mathrm{diam}(L) < m(r)$;
- $m(r) \searrow 0$ as $r \searrow 0$.

Of course, the existence of a modulus of local connectivity implies that X is locally connected. Conversely, every compact connected and locally connected set has a modulus of local connectivity.

We note that every modulus of local connectivity is also a modulus of path connectivity.

Proposition 5.4.9. *Let $m(r)$ be a modulus of local connectivity for a connected set $X \subset \mathbb{R}^2$. Let $x,y \in X$ such that $\mathrm{dist}(x,y) < r$. Then there exists a path ℓ between x and y with diameter at most $m(r)$.*

For the proof, see Proposition 2.2 of [23]. The following notion was introduced in [7].

Definition 5.4.10 (Carathéodory modulus). Let (W,w) be a pointed simply connected domain. A strictly increasing function $\eta : (0,a) \to \mathbb{R}$ is called a *Carathéodory modulus* if for every crosscut γ with $\mathrm{diam}(\gamma) < r < a$ we have $\mathrm{diam} N_\gamma < \eta(r)$.

We note (see, e.g., [7]) that this alternative modulus also allows one to characterize local connectivity.

Proposition 5.4.11. *There exists a Carathéodory modulus $\eta(r)$ such that $\eta(r) \searrow 0$ when $r \searrow 0$ if and only if the boundary ∂W is locally connected.*

5.4.5 Computable Carathéodory Theory

To simplify the exposition, we present the results for bounded domains only. However, all the theorems we formulate below may be stated for general simply connected domains on the Riemann sphere $\widehat{\mathbb{C}}$. In this case, the spherical metric on $\widehat{\mathbb{C}}$ would have to be used in the statements instead of the Euclidean one.

The following is shown in [7].

Theorem 5.4.12. *Suppose (W,w) is a bounded simply connected pointed domain. Suppose the Riemann mapping*

$$\phi : W \to \mathbb{D} \text{ with } \phi(w) = 0,\ \phi'(w) > 0$$

is computable. Then there exists an algorithm $\mathscr{A}$ which given a representation of a prime end $p \in \widehat{W}$ computes the value of $\hat{\phi}(p) \in S^1$.

In view of Theorem 5.3.1, we have the following.

Corollary 5.4.13. *Suppose we are given oracles for W as a lower-computable open set, for ∂W as a lower-computable closed set, and an oracle for the value of w as well. Given a representation of a prime end $p \in \widehat{W}$, the value $\hat{\phi}(p) \in S^1$ is uniformly computable.*

To state a "global" version of the above computability result, we use the structure of a computable metric space.

Theorem 5.4.14 ([7]). *In the presence of oracles for w and for ∂W, both the Carathéodory extension*

$$\hat{\phi} : \widehat{W} \to \overline{\mathbb{D}} \text{ and its inverse } \hat{f} \equiv \hat{\phi}^{-1} : \overline{\mathbb{D}} \to \widehat{W}$$

are computable, as functions between computable metric spaces.

Remark 5.4.15. The assumptions of Theorem 5.4.14 are stronger than those of Corollary 5.4.13: computability of ∂W implies lower-computability of W and ∂W, but not vice versa.

For the particular but important case when the domain W has a locally connected boundary, it is natural to ask what boundary information is required to make the extended map $\overline{f} : \overline{\mathbb{D}} \to \overline{W}$ computable. A natural candidate to consider is a description of the modulus of local connectivity. Indeed, as it was shown in [46, 47], a computable local connectivity modulus implies computability of the Carathéodory extension. Such a modulus of local connectivity, however, turned out to be unnecessary, as shown by the following two results proven in [7]. The first one says that the right boundary information to consider is the Carathéodory modulus $\eta(r)$, and the second one tells us that the two moduli, although classically equivalent, are indeed computationally different.

Theorem 5.4.16. *Suppose (W,w) is a pointed simply connected bounded domain with a locally connected boundary. Assume that the holomorphic bijection*

$$f : \mathbb{D} \to W \text{ with } f(0) = w,\ f'(0) > 0$$

is computable. Then the boundary extension

$$\overline{f} : \overline{\mathbb{D}} \to \overline{W}$$

is computable if and only if there exists a computable Carathéodory modulus $\eta(r)$ with $\eta(r) \searrow 0$ as $r \searrow 0$.

Remark 5.4.17. With routine modifications, the above result can be made *uniform* in the sense that there is an algorithm which from a description of f and η computes a description of $\overline{f}$, and there is an algorithm which from a description of $\overline{f}$ computes a Carathéodory modulus η. See for example [31] for statements made in this generality.

We note that the seemingly more "exotic" Carathéodory modulus cannot be replaced by the modulus of local connectivity in the above statement.

Theorem 5.4.18 ([7]). *There exists a simply connected domain W such that ∂W is locally connected and computable, and there exists a computable Carathéodory modulus $\eta(r) \searrow 0$, however, no computable modulus of local connectivity exists for ∂W.*

Finally, we turn to computational complexity questions in the cases when $\overline{\phi}$ or $\overline{f}$ are computable. Intuitively, when the boundary of the domain has a geometrically complex structure, one expects the Carathéodory extension to also be computationally complex. Using this idea, the following is shown in [7].

Theorem 5.4.19. *Let $q : \mathbb{N} \to \mathbb{N}$ be any computable function. There exist Jordan domains $W_1 \ni 0$, $W_2 \ni 0$ such that the following holds:*

- *the closures $\overline{W}_1$, $\overline{W}_2$ are computable;*
- *the extensions $\overline{\phi} : \overline{W}_1 \to \overline{\mathbb{D}}$ and $\overline{f} : \overline{\mathbb{D}} \to \overline{W}_2$ are both computable functions;*
- *the time complexity of $\overline{f}$ and $\overline{\phi}$ is bounded from below by $q(n)$ for large enough values of n.*

In other words, the computational complexity of the extended map can be arbitrarily high.

5.5 Computability in Complex Dynamics: Julia Sets

5.5.1 Basic Properties of Julia Sets

An excellent general reference for the material in this section is the textbook of Milnor [49].

The modern paradigm of numerical study of chaos (see, e.g., the article of J. Palis [51]) can be briefly summarized as follows. While the simulation of an individual orbit for an extended period of time does not make practical sense, one should study the limit set of a typical orbit (both as a spatial object and as a statistical distribution). One of the best-known illustrations of this approach is the numerical study of Julia sets in Complex Dynamics. The Julia set $J(R)$ is the *repeller* of a rational mapping R of degree $\deg R = d \geq 2$ considered as a dynamical system on the Riemann sphere

$$R : \widehat{\mathbb{C}} \to \widehat{\mathbb{C}},$$

that is, it is the attractor for (all but at most two) inverse orbits under R

$$z_0, z_{-1}, z_{-2}, \ldots, \text{ where } R(z_{-(n+1)}) = z_{-n}.$$

In fact, let δ_z denote the Dirac measure at $z \in \widehat{\mathbb{C}}$. Denote

$$\mu_n(z) = \frac{1}{d^n} \sum_{w \in R^{-n}(z)} \delta_w.$$

These probability measures assign equal weight to the n-th preimages of z, counted with multiplicity. Then for all points z except at most two, the measures $\mu_n(z)$ weakly converge to the *Brolin-Lyubich measure* λ of R, whose support is equal to $J(R)$ [17, 39].

Another way to define $J(R)$ is as the locus of chaotic dynamics of R, that is, the complement of the set where the dynamics is Lyapunov-stable.

Definition 5.5.1. Denote by $F(R)$ the set of points $z \in \widehat{\mathbb{C}}$ having an open neighborhood $U(z)$ on which the family of iterates $R^n|_{U(z)}$ is equicontinuous. The set $F(R)$ is called the Fatou set of R and its complement $J(R) = \widehat{\mathbb{C}} \setminus F(R)$ is the Julia set.

Finally, when the rational mapping is a polynomial

$$P(z) = a_0 + a_1 z + \cdots + a_d z^d : \mathbb{C} \to \mathbb{C}$$

an equivalent way of defining the Julia set is as follows. Obviously, there exists a neighborhood of ∞ on $\widehat{\mathbb{C}}$ on which the iterates of P uniformly converge to ∞. Denoting by $A(\infty)$ the maximal such domain of attraction of ∞ we have $A(\infty) \subset F(R)$. We then have

$$J(P) = \partial A(\infty).$$

The bounded set $\widehat{\mathbb{C}} \setminus A(\infty)$ is called *the filled Julia set*, and denoted $K(P)$; it consists of points whose orbits under P remain bounded:

$$K(P) = \{z \in \widehat{\mathbb{C}} | \sup_n |P^n(z)| < \infty\}.$$

Using the above definitions, it is not hard to make a connection between computability questions for polynomial Julia sets, and the general framework of computable

complex analysis discussed above. For instance, the Brolin-Lyubich measure λ on the Julia set is the *harmonic measure* at ∞ of the basin $A(\infty)$; that is, the pull-back of the Lebesgue measure on the unit circle by the appropriately normalized Riemann mapping of the basin $A(\infty)$. We will discuss the computability of this measure below. We will also see an even more direct connection to computability of a Riemann mapping when we talk about computability of polynomial Julia sets.

For future reference, let us summarize in a proposition below the main properties of Julia sets.

Proposition 5.5.2. *Let $R : \widehat{\mathbb{C}} \to \widehat{\mathbb{C}}$ be a rational function. Then the following properties hold:*

- *$J(R)$ is a non-empty compact subset of $\widehat{\mathbb{C}}$ and completely invariant: $R^{-1}(J(R)) = J(R)$;*
- *$J(R) = J(R^n)$ for all $n \in \mathbb{N}$;*
- *$J(R)$ has no isolated points;*
- *if $J(R)$ has non-empty interior, then it is the whole of $\widehat{\mathbb{C}}$;*
- *let $U \subset \widehat{\mathbb{C}}$ be any open set with $U \cap J(R) \neq \emptyset$. Then there exists $n \in \mathbb{N}$ such that $R^n(U) \supset J(R)$;*
- *periodic orbits of R are dense in $J(R)$.*

For a periodic point $z_0 = R^p(z_0)$ of period p its *multiplier* is the quantity $\lambda = \lambda(z_0) = DR^p(z_0)$. We may speak of the multiplier of a periodic cycle, as it is the same for all points in the cycle by the Chain Rule. In the case when $|\lambda| \neq 1$, the dynamics in a sufficiently small neighborhood of the cycle is governed by the Mean Value Theorem: when $|\lambda| < 1$, the cycle is *attracting* (*super-attracting* if $\lambda = 0$), if $|\lambda| > 1$ it is *repelling*. Both in the attracting and repelling cases, the dynamics can be locally linearized:

$$\psi(R^p(z)) = \lambda \cdot \psi(z) \tag{5.2}$$

where ψ is a conformal mapping of a small neighborhood of z_0 to a disk around 0.

In the case when $|\lambda| = 1$, so that $\lambda = e^{2\pi i\theta}$, $\theta \in \mathbb{R}$, the simplest to study is the *parabolic case* when $\theta = n/m \in \mathbb{Q}$, so λ is a root of unity. In this case R^p is not locally linearizable and $z_0 \in J(R)$.

In the complementary situation, two non-vacuous possibilities are considered: the *Cremer case*, when R^p is not linearizable, and the *Siegel case*, when it is. In the first case, the periodic point z_0 is called a *Cremer point*; in the second case it is called a *Siegel point*. In the Siegel case, the linearizing map ψ from (5.2) conjugates the dynamics of R^p on a neighborhood $U(z_0)$ to the irrational rotation by angle θ (the *rotation angle*) on a disk around the origin. The maximal such neighborhood of z_0 is called a *Siegel disk*.

A connected component H of the Fatou set is called a *Herman ring* if $f^p(H) = H$ for some $p \in \mathbb{N}$, and f^p restricted to H is conformally conjugate to an irrational rotation of a round annulus $\{r < |z| < 1\}$.

Fatou showed that for a rational mapping R with $\deg R = d \geq 2$ at most finitely many periodic orbits are non-repelling. A sharp bound on their number depending

on d has been established by Shishikura; it is equal to the number of critical points of R counted with multiplicity.

Fatou-Shishikura Bound. *For a rational mapping of degree d the number of non-repelling periodic cycles taken together with the number of cycles of Herman rings is at most $2d-2$. For a polynomial of degree d the number of non-repelling periodic cycles in $\mathbb{C}$ is at most $d-1$.*

Therefore, the last statement of Proposition 5.5.2 can be restated as follows.

- *repelling periodic orbits are dense in $J(R)$.*

To conclude the discussion of the basic properties of Julia sets, let us consider the simplest examples of non-linear rational endomorphisms of the Riemann sphere, the quadratic polynomials. Every affine conjugacy class of quadratic polynomials has a unique representative of the form $f_c(z) = z^2 + c$; the family

$$f_c(z) = z^2 + c,\ c \in \mathbb{C}$$

is often referred to as *the quadratic family*. For a quadratic map the structure of the Julia set is governed by the behavior of the orbit of the only finite critical point 0. In particular, the following dichotomy holds.

Proposition 5.5.3. *Let $K = K(f_c)$ denote the filled Julia set of f_c, and $J = J(f_c) = \partial K$. Then:*

- *$0 \in K$ implies that K is a connected, compact subset of the plane with connected complement, and $J = \partial K$ is also connected;*
- *$0 \notin K$ implies that $K = J$ is a planar Cantor set.*

The *Mandelbrot set* $\mathscr{M} \subset \mathbb{C}$ is defined as the set of parameter values c for which $J(f_c)$ is connected.

A rational mapping $R : \hat{\mathbb{C}} \to \hat{\mathbb{C}}$ is called *hyperbolic* if the orbit of every critical point of R is either periodic, or converges to an attracting cycle. As easily follows from the Implicit Function Theorem and considerations of the local dynamics of an attracting orbit, hyperbolicity is an open property in the parameter space of rational mappings of degree $d \geq 2$.

Considered as a rational mapping of the Riemann sphere, a quadratic polynomial $f_c(z)$ has two critical points: the origin, and the super-attracting fixed point at ∞. In the case when $c \notin \mathscr{M}$, the orbit of the former converges to the latter, and thus f_c is hyperbolic. A classical result of Fatou implies that whenever f_c has an attracting orbit in $\mathbb{C}$, this orbit attracts the orbit of the critical point. Hence, f_c is a hyperbolic mapping and $c \in \mathscr{M}$. The following conjecture is central to the field of dynamics in one complex variable.

Conjecture (Density of Hyperbolicity in the Quadratic Family. *Hyperbolic parameters are dense in $\mathscr{M}$.*

The Fatou-Shishikura Bound implies that a quadratic polynomial has at most one non-repelling cycle in the complex plane. Therefore, we will call the polynomial f_c (the parameter c, the Julia set J_c) *Siegel, Cremer,* or *parabolic* when it has an orbit of the corresponding type.

5.5.2 Occurrence of Siegel Disks and Cremer Points in the Quadratic Family

Before stating computability/complexity results for Julia sets we need to discuss in more detail the occurrence of Siegel disks in the quadratic family. For a number $\theta \in [0,1)$ denote by $[r_1, r_2, \dots, r_n, \dots]$, $r_i \in \mathbb{N} \cup \{\infty\}$ its possibly finite continued fraction expansion:

$$[r_1, r_2, \dots, r_n, \dots] \equiv \cfrac{1}{r_1 + \cfrac{1}{r_2 + \cfrac{1}{\cdots + \cfrac{1}{r_n + \cdots}}}} \tag{5.3}$$

Such an expansion is defined uniquely if and only if $\theta \notin \mathbb{Q}$. In this case, the *rational convergents* $p_n/q_n = [r_1, \dots, r_n]$ are the closest rational approximants of θ among the numbers with denominators not exceeding q_n.

Inductively define $\theta_1 = \theta$ and $\theta_{n+1} = \{1/\theta_n\}$. In this way,

$$\theta_n = [r_n, r_{n+1}, r_{n+2}, \dots].$$

We define the *Yoccoz's Brjuno function* as

$$\Phi(\theta) = \sum_{n=1}^{\infty} \theta_1 \theta_2 \cdots \theta_{n-1} \log \frac{1}{\theta_n}.$$

For an illustration see Figure 5.3. In 1972, Brjuno proved the following.

Theorem 5.5.4 ([16]). *Let R be an analytic map with a periodic point $z_0 \in \widehat{\mathbb{C}}$ of period p. Suppose the multiplier of the cycle*

$$\lambda = e^{2\pi i \theta} \text{ with } \Phi(\theta) < \infty,$$

then the local linearization equation (5.2) holds.

This theorem generalized the classical result by Siegel [59] which established the same statement for all Diophantine values of θ.

Note that a quadratic polynomial with a fixed Sigel disk with rotation angle θ after an affine change of coordinates can be written as

$$P_\theta(z) = z^2 + e^{2\pi i\theta} z. \tag{5.4}$$

For an example of a Julia set of a quadratic polynomial P_θ see Figure 5.2. In 1987

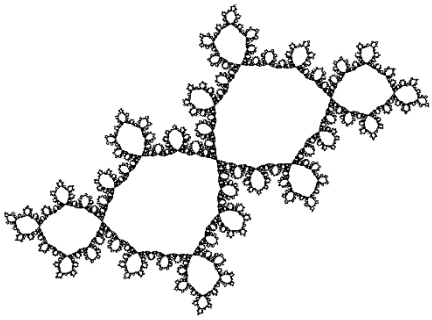


Fig. 5.2 The Julia set of P_θ for $\theta = [1,1,1,1,\ldots]$ (the inverse golden mean). This set is actually computable (see [13]).

Yoccoz [62] proved the following converse to Brjuno's Theorem.

Theorem 5.5.5 ([62]). *Suppose that for $\theta \in [0,1)$ the polynomial P_θ has a Siegel point at the origin. Then $\Phi(\theta) < \infty$.*

In fact, the value of the function Φ is directly related to the size of the Siegel disk in the following way.

Definition 5.5.6. Let $P(\theta)$ be a quadratic polynomial with a Siegel disk $\Delta_\theta \ni 0$. Consider a conformal isomorphism $\phi : \mathbb{D} \mapsto \Delta_\theta$ fixing 0. The *conformal radius of the Siegel disk Δ_θ* is the quantity

$$r(\theta) = |\phi'(0)|.$$

For all other $\theta \in [0,\infty)$ we set $r(\theta) = 0$.

By the Koebe One-Quarter Theorem of classical complex analysis, the internal radius of Δ_θ is at least $r(\theta)/4$. Yoccoz [62] has shown that the sum

$$\Phi(\theta) + \log r(\theta)$$

is bounded from below independently of $\theta \in \mathscr{B}$. Buff and Chéritat have greatly improved this result by showing the following.

Theorem 5.5.7 ([18]). *The function $\upsilon : \theta \mapsto \Phi(\theta) + \log r(\theta)$ extends to $\mathbb{R}$ as a 1-periodic continuous function.*

The following stronger conjecture exists (see [41]).

Marmi-Moussa-Yoccoz Conjecture ([41]). *The function* $\upsilon : \theta \mapsto \Phi(\theta) + \log r(\theta)$ *is Hölder of exponent* $1/2$.

It is important to note that computability of the function $\upsilon : \mathbb{R} \to \mathbb{R}$ would follow from the above conjecture. More generally, let us state the following (see [14]).

Theorem 5.5.8 (Conditional). *Suppose the function*

$$\upsilon : \theta \mapsto \Phi(\theta) + \log r(\theta)$$

has a computable modulus of continuity. Then it is uniformly computable on the entire interval $[0,1]$.

In view of the above, we formulate the following generalization of the Marmi-Moussa-Yoccoz Conjecture.

Conjecture 5.5.9. *The function*

$$\upsilon : \theta \mapsto \Phi(\theta) + \log r(\theta)$$

(see Figure 5.3) is uniformly computable on the interval $[0,1]$.

Lemma 5.5.10 (Conditional). *Suppose Conjecture 5.5.9 is true. Let* $\theta \in [0,1]$ *be such that* $\Phi(\theta)$ *is finite. Then there is an oracle Turing Machine* M_1^{ϕ} *computing* $\Phi(\theta)$ *with oracle access to* θ *if and only if there is an oracle Turing Machine* M_2^{ϕ} *computing* $r(\theta)$ *with oracle access to* θ.

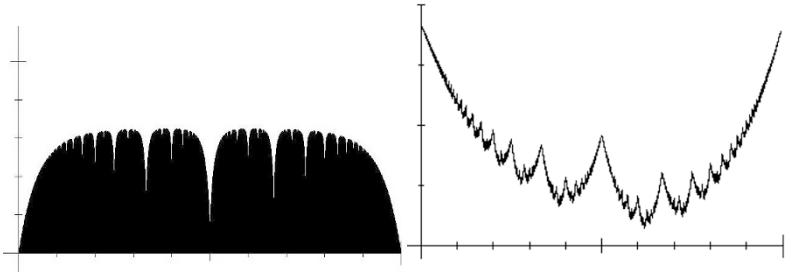


Fig. 5.3 The figure on the left is an attempt to visualize the function Φ by plotting the heights of $\exp(-\Phi(\theta))$ over a grid of Brjuno irrationals. On the right is the graph of the (conjecturally computable) function $\upsilon(x)$.
Both figures courtesy of Arnaud Chéritat

5.5.3 Computability of Julia Sets

When we address computability of a Julia set of a rational map R, it will always be by a Turing machine M^θ with oracle access to the coefficients of the rational map R. This corresponds to the numerical problem of drawing a Julia set when the coefficients of the map R are given with an arbitrary finite precision.

The question of computability of polynomial Julia sets was first raised in a paper of Zhong [63]. It was investigated exhaustively by Braverman and Yampolsky (see the monograph [14] as a general reference) with surprising results. For simplicity of exposition let us specialize to the case of the quadratic family.

Theorem 5.5.11 ([13]). *The filled Julia set K_c is always computable by an oracle Turing machine M^θ with an oracle for c.*

The special case of this theorem when K_c has empty interior was addressed in [5]. It is not hard to give an idea of the proof under this restriction. Indeed, the basin $A(\infty)$ is clearly lower-computable: simply take a large enough $R = R(c)$ such that for $|z| > R$, we have $|f_c(z)| > 2R$. Then the basin $A(\infty)$ is exhausted by the countable union of the computable (with an oracle for c) open sets

$$f_c^{-n}(\{|z| > R\}) \subset \widehat{\mathbb{C}}\,; n \in \mathbb{N}.$$

On the other hand, $K_c = J_c$ is a lower-computable closed set. It can be saturated by a countable sequence of computable (again with an oracle for c) finite sets

$$W_k \equiv \{\text{repelling periodic orbits of } f_c \text{ with period } \leq k\},\ k \in \mathbb{N}.$$

A simpler algorithm for lower-computing J_c, and something that is actually used in practice, is to find a single fixed point $p \in J_c$ (elementary considerations imply that from the two fixed points of f_c counted with multiplicity, at least one is either repelling or parabolic) and then saturate J_c by the sequence

$$W'_k \equiv \cup_{j \leq k} f_c^{-j}(p).$$

The proof of the general case of Theorem 5.5.11 is rather more involved.

The next statements will directly relate computability of Julia sets with computable Riemann mappings.

Theorem 5.5.12 ([12]). *Suppose f_c has a Siegel periodic point p and let $\Delta \ni p$ be the corresponding Siegel disk. Then J_c is computable by a Turing machine M^ϕ with an oracle for c if and only if the conformal radius of Δ is computable by a Turing machine with an oracle for c.*

Theorem 5.5.13 ([12]). *Suppose f_c has no Siegel points. Then the Julia set J_c is computable by a Turing machine M^ϕ with an oracle for c.*

Let us now specialize further to the case of the polynomials $P_\theta(z) = e^{2\pi i\theta} z + z^2$ with a neutral fixed point at the origin. The main result of Braverman and Yampolsky is the following.

Theorem 5.5.14 ([13]). *There exist computable values of θ (in fact, computable by an explicit, although very complicated, algorithm) such that J_θ is not computable.*

In view of Theorem 5.5.12, this is equivalent to the fact that the conformal radius r_θ is not computable. Assuming Conjecture 5.5.9, this would also be equivalent to the non-computability of the value of $\Phi(\theta)$. In fact, computable values of θ for which $\Phi(\theta)$ cannot be computed (unconditionally) can also be constructed (see [14]). In fact, assuming Conjecture 5.5.9, it is shown in [13] that θ in Theorem 5.5.14 can be poly-time.

We thus observe a surprising scenario: for above values of θ a finite inverse orbit of a point can be effectively, and possibly efficiently, computed with an arbitrary precision. Yet the repeller J_θ cannot be computed at all. This serves as a cautionary tale for applications of the numerical paradigm described above.

Fortunately, the phenomenon of non-computability of J_c is quite rare. Such values of c have Lebesgue measure zero. It is shown in [14] that assuming Conjecture 5.5.9 they have linear measure (Hausdorff measure with exponent 1) zero – a very meager set in the complex plane indeed. Furthermore, Conjecture 5.5.9 and some high-level theory of Diophantine approximations ([14]) imply that such values of c cannot be algebraic, so even if they are easy to compute, they are not easy to write down.

It is worth noting that in [15], Braverman and Yampolsky constructed computable values of θ such that the quadratic Julia sets J_θ are not computable and locally connected. In view of the above-discussed theory, for such maps, the basin of infinity $A(\infty) = \hat{\mathbb{C}} \setminus K_c$ is a natural example of a simply connected domain on the Riemann sphere with locally connected boundary such that the Riemann map is computable, but the Carathéodory extension is not (the boundary does not have a computable Carathéodory modulus).

5.5.4 Computational Complexity of Julia Sets

While the computability theory of polynomial Julia sets appears complete, the study of computational complexity of computable Julia sets offers many unanswered questions. Let us briefly describe the known results. As before, in all of them the Julia set of a rational function R is computed by a Turing Machine M^ϕ with an oracle for the coefficients of R.

The following theorem is independently due to Braverman [10] and Rettinger [55].

Theorem 5.5.15. *Every hyperbolic Julia set is poly-time.*

We note that the poly-time algorithm described in the above papers has been known to practitioners as *Milnor's Distance Estimator* [48]. Specializing again to the quadratic family f_c, we note that Distance Estimator becomes very slow (exp-time) for the values of c for which f_c has a parabolic periodic point. This would

appear to be a natural class of examples to look at for a lower complexity bound. However, surprisingly, Braverman [11] proved the following.

Theorem 5.5.16. *Parabolic quadratic Julia sets are poly-time.*

The algorithm presented in [11] is again explicit, and easy to implement in practice – it is a major improvement over Distance Estimator.

On the other hand, Binder, Braverman, and Yampolsky [4] proved the following.

Theorem 5.5.17. *There exist Siegel quadratics of the form $P_\theta(z) = e^{2\pi i\theta}z+z^2$ whose Julia set have an arbitrarily high time complexity.*

Given a lower complexity bound, such a θ can be produced constructively.

Another set of examples was recently constructed by Dudko and Yampolsky [28], who showed that there exist quadratic polynomials with Cremer points whose Julia sets have an arbitrarily high complexity. Cremer Julia sets are notoriously hard to draw in practice, and no high-resolution pictures have been produced to this day. An interesting possibility remains, however, that some of them may have low computational complexity, and be amenable to a clever algorithm, as the examples of [28] are quite restrictive.

Let us further specialize to real quadratic family f_c, $c \in \mathbb{R}$. In this case, the following was recently proved by Dudko and Yampolsky [27].

Theorem 5.5.18. *Almost every real quadratic Julia set is poly-time.*

This means that poly-time computability is a "physically natural" property in real dynamics. Conjecturally, the main technical result of [27] should imply the same statement for complex parameters c as well, but the conjecture in question (Collet-Eckmann parameters form a set of full measure among non-hyperbolic parameters) while long established, is stronger than the Density of Hyperbolicity Conjecture, and is currently out of reach.

It is also worth mentioning in this regard that most non-hyperbolic examples in real dynamics are infinitely renormalizable quadratic polynomials. The archetypal such example is the celebrated Feigenbaum polynomial. In a different paper, Dudko and Yampolsky [26] showed the following.

Theorem 5.5.19. *The Feigenbaum Julia set is poly-time.*

The above theorems raise the natural question of whether *all* real quadratic Julia sets are poly-time (the examples of [4] and [28] cannot have real values of c). That was recently ruled out by Rojas and Yampolsky [58].

Theorem 5.5.20. *There exist real quadratic Julia sets whose time complexity is arbitrarily high.*

5.5.5 Computing Julia Sets in Statistical Terms

As we have seen above, there are instances when for a rational map R, $d = \deg R \geq 2$, the set of limit points of the sequence of inverse images $R^{-n}(z)$ cannot be accurately simulated on a computer even in the "tame" case when $R \equiv f_c$ with a computable value of c. However, even in these cases we can ask whether the limiting *statistical distribution* of the points $R^{-n}(z)$ can be computed. As we noted above, for all $z \in \widehat{\mathbb{C}}$ except at most two, and every continuous test function ψ, the averages

$$\frac{1}{d^n} \sum_{w \in R^{-n}(z)} \psi(w) \underset{n\to\infty}{\longrightarrow} \int \psi \mathrm{d}\lambda,$$

where λ is the Brolin-Lyubich probability measure [17, 39] with $\mathrm{supp}(\lambda) = J(R)$. We can thus ask whether the value of the integral on the right-hand side can be algorithmically computed with arbitrary precision. Even if $J(R) = \mathrm{Supp}(\lambda)$ is not a computable set, the answer does not *a priori* have to be negative. Informally speaking, a positive answer would imply a dramatic difference between the rates of convergence in the following two limits:

$$\lim R^{-n}(z) \underset{\text{Hausdorff}}{\longrightarrow} J(R) \text{ and } \lim \frac{1}{d^n} \sum_{w \in R^{-n}(z)} \delta_w \underset{\text{weak}}{\longrightarrow} \lambda.$$

Indeed, the following was shown in [3].

Theorem 5.5.21. *The Brolin-Lyubich measure of R is always computable by a TM with an oracle for the coefficients of R.*

Even more surprisingly, the result of Theorem 5.5.21 is *uniform*, in the sense that there is a single algorithm that takes the rational map R as a parameter and computes the corresponding Brolin-Lyubich measure. Using the analytic tools given by the work of Dinh and Sibony [21], the authors of [3] also got the following complexity bound.

Theorem 5.5.22. *For each rational map R, there is an algorithm $\mathscr{A}(R)$ that computes the Brolin-Lyubich measure in exponential time, given an oracle for the coefficients of R.*

The running time of $\mathscr{A}(R)$ will be of the form $\exp(c(R) \cdot n)$, where n is the precision parameter, and $c(R)$ is a constant that depends only on the map R (but not on n). Theorems 5.5.21 and 5.5.22 are not directly comparable, since the latter bounds the growth of the computation's running time in terms of the precision parameter, while the former gives a single algorithm that works for all rational functions R.

As was pointed out above, the Brolin-Lyubich measure for a polynomial coincides with the harmonic measure of the complement of the filled Julia set. In view of Theorem 5.5.11, it is natural to ask what property of a computable compact set in the plane ensures computability of the harmonic measure of the complement.

We recall that a compact set $K \subset \widehat{\mathbb{C}}$ which contains at least two points is *uniformly perfect* if the moduli of the ring domains separating K are bounded from above. Equivalently, there exists some $C > 0$ such that for any $x \in K$ and $r > 0$, we have

$$(B(x,Cr) \setminus B(x,r)) \cap K = \emptyset \implies K \subset B(x,r).$$

In particular, every connected set is uniformly perfect. The following was shown in [3].

Theorem 5.5.23. *If a closed set $K \subset \mathbb{C}$ is computable and uniformly perfect, and has a connected complement, then the harmonic measure of the complement is computable.*

It is well known [40] that filled Julia sets are uniformly perfect. Theorem 5.5.23 thus implies Theorem 5.5.21 in the polynomial case. Computability of the set K is not enough to ensure computability of the harmonic measure: in [3] the authors presented a counterexample of a computable closed set whose complement has non-computable harmonic measure.

5.5.6 Applications of Computable Carathéodory Theory to Julia Sets: External Rays and Their Impressions

Informally (see [14] and [3] for a more detailed discussion), the parts of the Julia set J_c which are hard to compute are "inward pointing" decorations, forming narrow fjords of K_c. If the fjords are narrow enough, they will not appear in a finite-resolution image of K_c, which explains how the former can be computable even when J_c is not. Furthermore, a very small portion of the harmonic measure resides in the fjords, again explaining why it is always possible to compute the harmonic measure.

Suppose the Julia set J_c is connected, and denote by

$$\phi_c : \hat{\mathbb{C}} \setminus \overline{\mathbb{D}} \to \hat{\mathbb{C}} \setminus K_c$$

the unique conformal mapping satisfying the normalization $\phi_c(\infty) = \infty$ and $\phi_c'(\infty) = 1$. Carathéodory Theory (see, e.g., [49] for an exposition) implies that ϕ_c extends continuously to map the unit circle $\mathbb{S}^1$ onto the *Carathéodory completion* $\hat{J}_c$ of the Julia set. An element of the set $\hat{J}_c$ is a *prime end* p of $\mathbb{C} \setminus K_c$. The impression $\mathfrak{I}(p)$ of a prime end is a subset of J_c which should roughly be thought as a part of K_c accessible by a particular approach from the exterior. The harmonic measure ω_c can be viewed as the pushforward of the Lebesgue measure on $\mathbb{S}^1$ onto the set of prime end impressions.

In view of the above-quoted results, from the point of view of computability, prime end impressions should be seen as borderline objects. On the one hand, they are subsets of the Julia set which may be non-computable; on the other they are

"visible from infinity", and as we have seen accessibility from infinity generally implies computability.

It is thus natural to ask whether the impression of a prime end of $\hat{\mathbb{C}} \setminus K_c$ is always computable by a TM with an oracle for c. To formalize the above question, we need to describe a way of specifying a prime end. We recall that *the external ray* R_α of angle $\alpha \in \mathbb{R}/\mathbb{Z}$ is the image under ϕ_c of the radial line $\{re^{2\pi i\alpha} : r > 1\}$. The curve

$$R_\alpha = \phi_c(\{re^{2\pi i\alpha} : r > 1\})$$

lies in $\hat{\mathbb{C}} \setminus K_c$. The *principal impression of an external ray* $\mathfrak{P}(R_\alpha)$ is the set of limit points of $\phi_c(re^{2\pi i\alpha})$ as $r \to 1$. If the principal impression of R_α is a single point z, we say that R_α *lands* at z. External rays play a very important role in the study of polynomial dynamics.

It is evident that every principal impression is contained in the impression of a unique prime end. We call the impression of this prime end the *prime end impression of an external ray* and denote it $\mathfrak{I}(R_\alpha)$. A natural refinement of the first question is the following: suppose α is a computable angle; is the prime end impression $\mathfrak{I}(R_\alpha)$ computable? As was shown in [8], the answer is emphatically negative.

Theorem 5.5.24. *There exists a computable complex parameter c and a computable Cantor set of angles $C \subset \mathbb{S}^1$ such that for every angle $\alpha \in C$, the impression $\mathfrak{I}(R_\alpha) \subset J_c$ is not computable. Moreover, any compact subset $K \Subset J_c$ which contains $\mathfrak{I}(R_\alpha)$ is non-computable.*

This statement illustrates yet again how subtle, and frequently counter-intuitive, the answers to natural computability questions may be when it comes to Julia sets, and, by extension, to other fractal invariant sets in low-dimensional dynamics.

5.5.7 On the Computability of the Mandelbrot Set

Let us recall that the Mandelbrot set $\mathscr{M}$ is defined to be the *connectedness locus* of the family $f_c(z) = z^2 + c$, $c \in \mathbb{C}$: the set of complex parameters c for which the Julia set $J(f_c)$ is connected. The boundary of $\mathscr{M}$ corresponds to the parameters near which the geometry of the Julia set undergoes a dramatic change. For this reason, $\partial\mathscr{M}$ is referred to as the *bifurcation locus*. As already discussed in Section 5.5.1, $J(f_c)$ is connected precisely when the critical point 0 does not escape to infinity. Therefore, $\mathscr{M}$ can be equivalently defined as

$$\mathscr{M} = \{c \in \mathbb{C} : \sup_{n\in\mathbb{N}} |f_c^n(0)| < \infty\}.$$

$\mathscr{M}$ is widely known for the spectacular beauty of its fractal structure, and an enormous amount of effort has been made in order to understand its topological and geometrical properties. It is easy to see that $\mathscr{M}$ is a compact set, equal to the closure of its interior, and contained in the disk of radius 2 centered at the origin. Douady

and Hubbard have shown [24] that $\mathscr{M}$ is connected and simply connected. In this section we will discuss the computability properties of $\mathscr{M}$, a question raised by Penrose in [52]. Hertling showed the following in [32].

Theorem 5.5.25. *The complement of the Mandelbrot set is a lower-computable open set, and its boundary $\partial\mathscr{M}$ is a lower-computable closed set.*

Proof. Note that

$$\mathbb{C} \setminus \mathscr{M} = \bigcup_{n\in\mathbb{N}} \{c \in \mathbb{C} : |f_c^n(0)| > 2\}.$$

Since $f_c^n(0)$ is computable as a function of c, uniformly in n, it follows that the open sets $\{c \in \mathbb{C} : |f_c^n(0)| > 2\}$ are uniformly lower-computable, which proves the first claim. For the second claim, a simple way to compute a dense sequence of points in $\partial\mathscr{M}$ is by computing the so-called *Misiurewicz parameters*, for which the critical point of f_c is strictly pre-periodic. In other words, the union of sets

$$W_{l,p} = \{c \mid f_c^i(0) \neq 0 \text{ if } i \leq l+p \text{ and } f_c^{l+p}(0) = f_c^l(0)\} \text{ for } l, p \in \mathbb{N}$$

is dense in $\partial\mathscr{M}$.

In virtue of Theorem 5.3.1, we note the following.

Corollary 5.5.26. *The Riemann map*

$$\phi : \hat{\mathbb{C}} \setminus \overline{\mathbb{D}} \to \hat{\mathbb{C}} \setminus \mathscr{M},$$

sending the complement of the unit disk to the complement of the Mandelbrot set, is computable.

It is unknown whether the whole of $\mathscr{M}$ is a lower-computable set. A positive answer would imply computability of $\mathscr{M}$. In fact, Hertling [32] also showed the following.

Theorem 5.5.27. *The Density of Hyperbolicity Conjecture implies that $\mathscr{M}$ is lower-computable, and hence, computable.*

Indeed, let U_m be defined as the set of parameters $c \in \mathbb{C}$ such that f_c has a point $p \in \mathbb{C}$ with $f_c^m(p) = p$ and $|Df_c^m(p)| < 1$. It is not hard to see that such a set is lower-computable. Density of Hyperbolicity implies that the sets U_m are dense in $\mathscr{M}$.

Note that the same statements hold for the $n-1$ complex dimensional connectedness locus of the family of polynomials of degree n, with similar proofs. On the other hand, it is possible to construct a one-parameter complex family of polynomials in which the corresponding objects are not computable. In [20], Coronel, Rojas, and Yampolsky have recently shown the following.

Theorem 5.5.28. *There exists an explicitly computable complex number λ such that the bifurcation locus of the one-parameter family*

$$f_c(z) = \lambda z + cz^2 + z^3$$

is not computable.

An interesting related question is the computability of the area of $\mathscr{M}$. Since $\mathbb{C} \setminus \mathscr{M}$ is a lower-computable set, it follows that the area of $\mathscr{M}$ is an upper-computable real number. The following simple conditional implication holds.

Proposition 5.5.29. *If the area of $\mathscr{M}$ is computable, then $\mathscr{M}$ is computable.*

Proof. Since $\mathbb{C} \setminus \mathscr{M}$ is a lower-computable set, if the area of $\mathscr{M}$ were a computable real number, by Proposition 2.3.1.1 from [57] the Lebesgue measure restricted to $\mathscr{M}$ would be a computable measure. By Proposition 5.2.19, the support of this measure is lower-computable. Since the support in this case is $\mathscr{M}$, the result follows.

Let us also note another famous conjecture about the Mandelbrot set.

Conjecture (MLC). *The Mandelbrot set $\mathscr{M}$ is locally connected.*

MLC is known (see [22]) to be stronger than the Density of Hyperbolicity Conjecture, and thus also implies computability of $\mathscr{M}$ and of $\partial\mathscr{M}$. In this case, by virtue of Corollary 5.5.26 and Theorem 5.4.16, it would be natural to ask whether $\mathscr{M}$ admits a computable Carathéodory modulus. Some partial results in this direction are known (see, e.g., [34]). In all of them, local connectedness is established at subsets of points of $\mathscr{M}$ by providing a constructive Carathéodory modulus at these points.

References

1. Banach, S., Mazur, S.: Sur les fonctions calculables. Annales de la Société Polonaise de Mathématique **16**, 219 (1937)
2. Binder, I., Braverman, M.: Derandomization of euclidean random walks. In: M. Charikar, K. Jansen, O. Reingold, J.D.P. Rolim (eds.) Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, pp. 353–365. Springer Berlin Heidelberg (2007)
3. Binder, I., Braverman, M., Rojas, C., Yampolsky, M.: Computability of Brolin-Lyubich measure. Comm. Math. Phys. **308**(3), 743–771 (2011)
4. Binder, I., Braverman, M., Yampolsky, M.: On computational complexity of Siegel Julia sets. Comm. Math. Phys. **264**(2), 317–334 (2006)
5. Binder, I., Braverman, M., Yampolsky, M.: Filled Julia sets with empty interior are computable. Foundations of Computational Mathematics **7**(4), 405–416 (2007)
6. Binder, I., Braverman, M., Yampolsky, M.: On the computational complexity of the Riemann mapping. Arkiv för Matematik **45**(2), 221–239 (2007)
7. Binder, I., Rojas, C., Yampolsky, M.: Computable Carathéodory theory. Adv. Math. **265**, 280–312 (2014)
8. Binder, I., Rojas, C., Yampolsky, M.: Non-computable impressions of computable external rays of quadratic polynomials. Communications in Mathematical Physics **335**(2), 739–757 (2015)
9. Bishop, E., Bridges, D.S.: Constructive Analysis, *Grundlehren der Mathematischen Wissenschaften*, vol. 279. Springer, Berlin (1985)

10. Braverman, M.: Computational complexity of Euclidean sets: Hyperbolic Julia sets are polytime computable. Master's thesis, University of Toronto (2004)
11. Braverman, M.: Parabolic Julia sets are polynomial time computable. Nonlinearity **19**(6), 1383–1401 (2006)
12. Braverman, M., Yampolsky, M.: Non-computable Julia sets. Journal of the American Mathematical Society **19**(3), 551–578 (2006)
13. Braverman, M., Yampolsky, M.: Computability of Julia sets. Moscow Mathematical Journal **8**(2), 185–231, 399 (2008)
14. Braverman, M., Yampolsky, M.: Computability of Julia Sets, *Algorithms and Computation in Mathematics*, vol. 23. Springer, Berlin (2008)
15. Braverman, M., Yampolsky, M.: Constructing locally connected non-computable Julia sets. Communications in Mathematical Physics **291**(2), 513–532 (2009)
16. Brjuno, A.D.: Analytic forms of differential equations. Trans. Mosc. Math. Soc **25**, 119–262 (1971)
17. Brolin, H.: Invariant sets under iteration of rational functions. Ark. Mat. **6**, 103–144 (1965)
18. Buff, X., Chéritat, A.: The Brjuno function continuously estimates the size of quadratic Siegel disks. Annals of Math. **164**(1), 265–312 (2006)
19. Cheng, H.: A constructive Riemann mapping theorem. Pacific J. Math. **44**, 435–454 (1973)
20. Coronel, D., Rojas, C., Yampolsky, M.: Non computable Mandelbrot-like sets for a one-parameter complex family. Inform. and Comput. **262**(part 1), 110–122 (2018)
21. Dinh, T., Sibony, N.: Equidistribution speed for endomorphisms of projective spaces. Math. Ann. **347**, 613–626 (2009)
22. Douady, A.: Descriptions of compact sets in **C**. In: L.R. Goldberg (ed.) Topological Methods in Modern Mathematics (Stony Brook, NY, 1991), pp. 429–465. Publish or Perish, Houston, TX (1993)
23. Douady, A., Hubbard, J.H.: Exploring the Mandelbrot set. The Orsay notes, English version of [25]
24. Douady, A., Hubbard, J.H.: Itération des polynômes quadratiques complexes. CR Acad. Sci. Paris **294**, 123–126 (1982)
25. Douady, A., Hubbard, J.H.: Étude dynamique de polynômes complexes. *Documents Mathématiques*. Société Mathématique de France (2007)
26. Dudko, A., Yampolsky, M.: Poly-time computability of the Feigenbaum Julia set. Ergodic Theory and Dynamical Systems **36**(8), 2441–2462 (2016)
27. Dudko, A., Yampolsky, M.: Almost every real quadratic polynomial has a poly-time computable Julia set. Found. Comput. Math. **18**(5), 1233–1243 (2018)
28. Dudko, A., Yampolsky, M.: On computational complexity of Cremer Julia sets. Fund. Math. **252**(3), 343–353 (2021)
29. Galatolo, S., Hoyrup, M., Rojas, C.: Dynamics and abstract computability: computing invariant measures. Discrete and Continuous Dynamical Systems. Series A **29**(1), 193–212 (2011)
30. Grzegorczyk, A.: Computable functionals. Fundamenta Mathematicae **42**, 168–202 (1955)
31. Hertling, P.: An effective Riemann Mapping Theorem. Theoretical Computer Science **219**, 225–265 (1999)
32. Hertling, P.: Is the Mandelbrot set computable? Mathematical Logic Quarterly **51**(1), 5–18 (2005)
33. Hoyrup, M., Rojas, C.: Computability of probability measures and Martin-Löf randomness over metric spaces. Information and Computation **207**(7), 830–847 (2009)
34. Hubbard, J.H.: Local connectivity of Julia sets and bifurcation loci: three theorems of J.-C. Yoccoz. In: L.R. Goldberg (ed.) Topological Methods in Modern Mathematics (Stony Brook, NY, 1991), pp. 467–511. Publish or Perish, Houston, TX (1993)
35. Ko, K.I.: Complexity Theory of Real Functions. *Progress in Theoretical Computer Science*. Birkhäuser, Boston (1991)
36. Koebe, P.: Über eine neue Methode der konformen Abbildung und Uniformisierung. Nachr. Ges. Wiss. Göttingen, Math.-Phys. Kl. **1912**, 844–848 (1912)
37. Kühnau, R.: Numerische Realisierung konformer Abbildungen durch "Interpolation". Z. Angew. Math. Mech. **63**(12), 631–637 (1983)

38. Lacombe, D.: Extension de la notion de fonction récursive aux fonctions d'une ou plusieurs variables réelles I. Comptes Rendus Académie des Sciences Paris **240**, 2478–2480 (1955)
39. Lyubich, M.: The measure of maximal entropy of a rational endomorphism of a Riemann sphere. Funktsional. Anal. i Prilozhen. **16**, 78–79 (1982)
40. Manẽ, R., da Rocha, L.: Julia sets are uniformly perfect. Proc. Amer. Math. Soc. **116**, 251–257 (1992)
41. Marmi, S., Moussa, P., Yoccoz, J.C.: The Brjuno functions and their regularity properties. Commun. Math. Phys. **186**, 265–293 (1997)
42. Marshall, D.E.: Numerical conformal mapping software: Zipper, available from http://www.math.washington.edu/∼marshall/zipper.html
43. Marshall, D.E., Rohde, S.: Convergence of a variant of the Zipper algorithm for conformal mapping. SIAM J. Numer. Anal. **45**(6), 2577–2609 (2007)
44. Mazur, S.: Computable Analysis, *Razprawy Matematyczne*, vol. 33. Pánstwowe Wydawnictwo Naukowe, Warsaw (1963)
45. Mazurkiewicz, S.: Über die Definition der Primenden. Fund. Math. **26**(1), 272–279 (1936)
46. McNicholl, T.H.: An effective Carathéodory theorem. Theory of Computing Systems **50**(4), 579–588 (2012)
47. McNicholl, T.H.: Computing boundary extensions of conformal maps. LMS Journal of Computation and Mathematics **17**(1), 360–378 (2014)
48. Milnor, J.: Self-similarity and hairiness in the Mandelbrot set. In: M. Tangora (ed.) Computers in Geometry and Topology, *Lect. Notes Pure Appl. Math.*, vol. 114, pp. 211–257. Marcel Dekker (1989)
49. Milnor, J.: Dynamics in One Complex Variable. Introductory Lectures, 3rd edn. Princeton University Press (2006)
50. Osgood, W.F.: On the existence of the Green's function for the most general simply connected plane region. Trans. Amer. Math. Soc. **1**(3), 310–314 (1900)
51. Palis, J.: A global view of dynamics and a conjecture on the denseness of finitude of attractors. Astérisque **261**, 339 – 351 (2000)
52. Penrose, R.: The emperor's new mind. RSA Journal **139**(5420), 506–514 (1991)
53. Pommerenke, C.: Univalent functions, *Studia Mathematica/Mathematische Lehrbücher*, vol. XXV. Vandenhoeck & Ruprecht (1975)
54. Post, E.L.: Finite combinatory processes - Formulation 1. The Journal of Symbolic Logic **1**, 103–105 (1936)
55. Rettinger, R.: A fast algorithm for Julia sets of hyperbolic rational functions. In: V. Brattka, L. Staiger, K. Weihrauch (eds.) Proceedings of the 6th Workshop on Computability and Complexity in Analysis, *Electronic Notes in Theoretical Computer Science*, vol. 120, pp. 145–157. Elsevier, Amsterdam (2005), 6th International Workshop, CCA 2004, Wittenberg, Germany, August 16–20, 2004
56. Rettinger, R.: Computability and complexity aspects of univariate complex analysis. Habilitation thesis. FernUniversität Hagen (2008)
57. Rojas, C.: Randomness and ergodic theory: an algorithmic point of view. Ph.D. thesis. École Polytechnique and Università di Pisa (2008)
58. Rojas, C., Yampolsky, M.: Real quadratic Julia set can have arbitrarily high complexity. Foundations of Computational Mathematics (2020)
59. Siegel, C.: Iteration of analytic functions. Ann. of Math. **43**(2), 607–612 (1942)
60. Turing, A.M.: On computable numbers, with an application to the "Entscheidungsproblem". Proceedings of the London Mathematical Society **42**(2), 230–265 (1937)
61. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)
62. Yoccoz, J.C.: Petits diviseurs en dimension 1. S.M.F., Astérisque **231** (1995)
63. Zhong, N.: Recursively enumerable subsets of R^q in two computing models: Blum-Shub-Smale machine and Turing machine. Theoretical Computer Science **197**, 79–94 (1998)
64. Zhou, Q.: Computable real-valued functions on recursive open and closed subsets of Euclidean space. Mathematical Logic Quarterly **42**, 379–409 (1996)

Chapter 6
A Survey on Analog Models of Computation

Olivier Bournez and Amaury Pouly

Abstract We present a survey on analog models of computation. Analog can be understood both as computing by analogy, or as working on the continuum. We consider both approaches, often intertwined, with a point of view mostly oriented by computation theory.

6.1 Introduction

There is a clear ambiguity about the sense of the word *analog* when talking about analog computations. Nowadays, it is often understood as being the opposite of *digital*: the former is working on continuous quantities, while the latter is working over discrete values, typically bits or words. However, historically, *analog computation* got its name from computation by analogy, i.e., by systems built in such a way that they evolve in exactly the same way as the system they were intended to model or simulate [240, 326]. These two understandings are orthogonal and various machines analog in both or one and not the other sense have been conceived [326]. Notice also that even *discrete* versus *continuous* is not a clear dichotomy, and most of the analog machines that were historically built were actually hybrid [326].

We will mainly focus on models of computation, with a point of view possibly oriented by computation theory (computability, complexity, models of computation).

All considered models can be described as particular dynamical systems: a dynamical system is mathematically defined as the action of a subgroup $\mathscr{T}$ of $\mathbb{R}$ on

Olivier Bournez
École Polytechnique, Palaiseau, France
e-mail: bournez@lix.polytechnique.fr

Amaury Pouly
Max Planck Institute for Software Systems, Saarbrücken, Germany
e-mail: pamaury@mpi-sws.org

V. Brattka, P. Hertling (eds.), *Handbook of Computability and Complexity in Analysis*,
Theory and Applications of Computability, https://doi.org/10.1007/978-3-030-59234-9_6

a space X, i.e., by a function (a *flow*) $\phi : \mathscr{T} \times X \to X$ satisfying the following two equations:

$$\phi(0,x) = x \tag{6.1}$$

$$\phi(t,\phi(s,x)) = \phi(t+s,x). \tag{6.2}$$

The function $\phi(t,x)$ is intended to give the position at time t of the system if started at position x at time 0, and the above equations simply express expected properties.

Subgroups $\mathscr{T}$ of $\mathbb{R}$ are known to be either dense in $\mathbb{R}$ or isomorphic to the integers. In the first case, the time is said to be *continuous*, in the latter case, *discrete*.

A dynamical system is often alternatively described by some space X and some function $f : X \to X$. Indeed, in the discrete-time case, giving ϕ is equivalent to giving $f : X \to X$ with $f(x) = \phi(1,x)$: the trajectory starting from some state x_0 then corresponds to the iterations of f on x_0 from Equation (6.2). In the continuous-time case, not all dynamical systems correspond to differential equations, but as soon as ϕ is continuously differentiable, a case covering a very wide class of systems in practice, we can write $y' = f(y)$ where $f(y) = \frac{\mathrm{d}}{\mathrm{d}t}\phi(t,y)\big|_{t=0}$. Giving ϕ is then also equivalent to giving a function $f : X \to X$: the trajectory starting from some state x_0 corresponds to the solution of the Initial Value Problem (IVP) $y' = f(y), y(0) = x_0$.

We can then classify models according to their space X: we will mainly focus in this chapter on the case where the space X involves real numbers, i.e., is *continuous*. Typically $X = \mathbb{R}^n$ or $X = \mathbb{R}^n \times \mathbb{N}^k$ or can be encoded naturally in similar spaces. We will say that such a space is *continuous*, in contrast to spaces like $\mathbb{N}^k$ (or the set of words over a given alphabet, or the set of configurations of a model such as a Turing machine), which would correspond to a *discrete* space. Discrete time and space corresponds to classical computability and complexity and we do not intend to cover them in this chapter. We will nevertheless consider some unconventional discrete-time and -space models, such as population protocols or chemical reaction networks, as they are unconventional and turn out to be closely related to analog models as shown by various recent results.

A classification of some of the models considered in this chapter according to these discrete/continuous time/space dichotomies is provided by Figure 6.1. This classification is not perfect and is debatable. For example, cellular automata (which we will consider later on as a spatial model) are here considered as evolving over a discrete space. But they may also be considered as evolving over $\{0,1\}^{\mathbb{N}}$ (i.e., Baire's space), and we agree that there is no fundamental difference between $\mathbb{R}$ and $\{0,1\}^{\mathbb{N}}$. As another example, the discrete-space / continuous-time quadrant is rather empty, but we agree that many models have an underlying semantics that can be expressed in the language of continuous-time Markov chains, and for example stochastic chemical reaction networks could also fall in this quadrant. Actually, many models come indeed in various flavors and could be turned into various quadrants: for example, asynchronous cellular automata can arguably be considered as being continuous-time and discrete-space, or continuous-time and continuous-space.

Space / Time	Discrete	Continuous
Discrete	Turing machines Lambda calculus Recursive functions Post systems Cellular automata Stack automata Finite state automata Population protocols (Section 6.2.8) Chemical reaction networks (Section 6.6.5) Petri nets ⋮	Neural networks (Section 6.2.3) Deep learning models (Section 6.2.3) Blum-Shub-Smale machines (Section 6.2.5) Hybrid systems (Section 6.2.4) Natural computing influence dynamics (Section 6.2.6) Signal machines (Section 6.2.10.3) Continuous automata (Section 6.5.3) ⋮
Continuous	Boolean difference equation models	Shannon's GPACs (Section 6.2.2.1) Hopfield's neural networks (Section 6.6.1) Physarum computing (Section 6.2.6.2) Reaction-diffusion systems (Section 6.2.11) Hybrid systems (Section 6.2.4) Timed automata (Section 6.2.4.1) Large-population protocols (Section 6.2.8.1) Black hole models (Section 6.2.9) Computational fields (Section 6.2.10.1) $\mathbb{R}$-recursive functions (Section 6.5.2) Spiking neuron models (Section 6.2.3) ⋮

Fig. 6.1 A tentative classification of some computational models, according to their space and time.

Notice that many quantum models of computation can also be considered possibly as analog and would fit this description: as we have to make choices, we decided that they are out of the scope of the current chapter. Notice that analog models, in particular models in the continuum, can be seen as fitting in some way in between classical and quantum models. Several results about quantum models are deeply depend upon the fact that the underlying space is the field $\mathbb{C}$ of complex numbers, while analog models can sometimes emulate some of the constructions by working over $\mathbb{R}^2$. Similarly, computations with quantum models sometimes assume measurements to be possible at unit cost, while this is not considered to be possible in most analog models. Discussing analog models in this spirit helps to distinguish the aspects which are closely coupled to the quantum world from others.

Since most of the other chapters of this book are dedicated to the computable analysis framework, we will intentionally not discuss this approach in the current chapter.

Previous surveys on the field of analog computation include Pekka Orponen's [267], Olivier Bournez and Manuel L. Campagnolo's [64] focusing on continuous-time computation, and Bruce J. MacLennan's chapter on "Analog Computation" of the *Encyclopedia of Complexity and Systems Science.* [240]. A very instructive book about the history of analog machines has recently been published [326]. There is also an extensive literature, mostly forgotten today, about analog machines dating to the times where most machine programming was analog.

6.2 Various Analog Machines and Models

6.2.1 Historical Accounts

As we said, historically, analog computation was mainly referring to computation by analogy. Very instructive quotations supporting this claim can be found in [326] and [240].

Actually, several historical computers presented as among the first *digital* computers ever built turn out to be primarily analog in the above sense. This includes ENIAC, which interestingly stands for *"Electronic Numerical Integrator and Computer"*. ENIAC is often said to be *"programmed"* but the term *"wired"* would be closer to reality, since this system mainly consisted of a large collection of arithmetic machines.

It may also help to understand that the term *computer* historically referred to a person who carried out calculations or computations. Only from the middle of the twentieth century, did the word start to refer to a *machine* that carries out computations. Determining which systems can actually be considered to be computers is a very intriguing question, related to deep philosophical questions out of the scope of the present chapter.

However, we list here some of the first machines ever built that can be classified as analog: this includes Blaise Pascal's 1642 *Pascaline*, as well as Johann Martin Hermann's 1814 *Planimeter* (a simple device to compute surfaces based on Green's theorem), and Bill Phillips's 1949 *MONIAC* (*Monetary National Income Analogue Computer*, a machine that was using fluidic logic to model the behavior of an economy). The *Antikythera mechanism*, discovered close to the Greek island of Antikythera in 1901, dated from earlier than 87 BC, whose purpose was to predict astronomical positions and phenomena, is also often considered to be an analog computer. However, even if these machines were clearly computing various quantities or data and may or may not be called computers, we admit that even the question of whether such machines can be classified as *analog* is debatable. Indeed, many ar-

ticles or books consider them as such nowadays (see e.g. [326]), but these statements conflict with the literature in the history of computing from historians.

However, with no contest, the first truly programmable (*"general-purpose"* using Shannon's 1941 terminology) analog computer was Vannevar's Bush 1931 *Differential Analyzer*, which is the topic of the next section due to its historical and fundamental importance.

Further detailed historical accounts about analog machines and computations can be found in the recent monograph [326]. Its author, Bernd Ulmann, also maintains an informative web site with instructive pictures and videos [1].

The history of analog computation and devices is also discussed by historians. For general literature on the history of analog computers, we refer to [109], [249] or [333]. For references related to history of analog instruments, see [168], [167]. The history of the Differential Analyzer is discussed in [269]. See also [198] and [197] for national accounts on the developments of differential analyzers. Historical accounts on the ENIAC can be found in [195]. Recently published accounts also include the monograph [290], and the analysis of the work of Douglas R. Hartree [157] and Charles Babbage [154] and of particular devices or techniques [155, 156] before differential analyzers.

6.2.2 Differential Analyzers

Probably the best-known universal continuous-time machine is the *Differential Analyzer (DA)*, built for the first time in 1931 at MIT under the supervision of Vannevar Bush [90]. The idea of assembling integrator devices to solve differential equations dates back to Lord Kelvin in 1876 [321]. Kelvin was looking for a faster way to compute the harmonics of a function using its Fourier transform, with applications to tidal and meteorological observations. He came up with the idea of using a rotating disc-cylinder-globe system to compute the integral of a product: this is essentially the fundamental operation of the Differential Analyzer, although it took over 50 years to solve the mechanical problems involved in this machine.

The first DAs were entirely mechanical, and later became electronic: see [80, 326]. Their primary purpose was to solve differential equations, especially the ones coming from problems in engineering. By the 1960s, differential analyzers were progressively discarded in favor of digital technology. Many accounts on the history and applications of these machines can be found in [326].

6.2.2.1 A Mathematical Model of the Differential Analyzer: the GPAC

The *General Purpose Analog Computer* (*GPAC*) was introduced by Shannon in 1941 as a mathematical idealization of the Differential Analyzer [303], while he was working on this machine at MIT to get money for his studies. A GPAC is basically a circuit made up of a finite number of units, described in Figure 6.2, which are

interconnected by wires, possibly with loops (retroactions) on some of the wires. A function $f : \mathbb{R} \to \mathbb{R}$ is said to be *generated* if it corresponds to the value read on some of the wires of the circuit as a function of time.

The model was later refined in a series of papers [284, 227, 186, 184]. Indeed, a GPAC circuit may not define a unique function (it could have no or several solutions), which is problematic. An arguably more modern presentation of the GPAC is to use ordinary differential equations instead of circuits. The equivalence of GPAC circuits with polynomial ordinary differential equations will be discussed in Section 6.7.1.

Rubel also introduced the *Extended Analog Computer* (*EAC*) as an extension of the GPAC [293]. The EAC features a broader class of operations such as partial differentiation and restricted limits, allowing it to solve boundary value problems. Rubel's motivation for introducing the EAC was that the GPAC was "a limited machine". This claim has been somewhat changed since then (see Section 6.7.1) and the EAC has seen few theoretical developments. Although Rubel envisioned the EAC as a "purely conceptual machine", there is ongoing research to implement it, with some currently working prototypes [252, 254, 253].

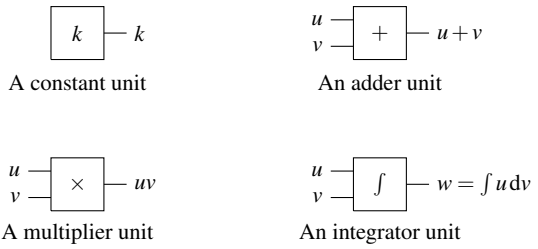


Fig. 6.2 Basic units used in a GPAC circuit.

6.2.3 Neural Networks and Deep Learning Models

In the 1980's and 1990's, *artificial neural networks* gave birth to a renewal of interest in analog computations. This enthusiasm declined until recently when the success of *deep learning* in several impressive applications in various fields of artificial intelligence, such as speech recognition, image recognition and natural language processing emerged. The first machine able to beat all the best professional *Go* players is based on deep learning technology.

Current deep learning models may have a rather complex architecture built from various modules, but most of these modules are essentially artificial neural network models. An artificial neural network (ANN) consists of many simple, connected processors called *neurons*. The state of each neuron is given by some real num-

ber called its activation value. Designated *input neurons* get their activation values from the environment. All other neurons evolve by applying a composition of a certain one-variable function σ (usually a sigmoid) with an affine combination of the activations of the neurons to which they are connected. Finally, specific neurons, considered as *output neurons*, may trigger actions on the environment: see [54] for an overview.

Most of the work related to (deep) artificial neural networks is nowadays devoted to finding architectures and weights that make an ANN exhibit a desired behavior in a given context of application A popular library to describe corresponding architectures is *TensorFlow* [2]. This is basically a library to describe such architectures, and hence can be considered in many aspects as a library for particular analog computations.

A very popular and successful method to determine suitable weights for solving a given problem is *backpropagation*, which is a reinforcement learning technique based on a gradient descent method applied to an error function expected to be minimized over the learning set, see [54].

Applying any gradient descent method requires a differentiable error function with respect to all involved parameters. The very large number of applications of deep learning techniques have recently led to the emergence of various other analog models of computation. All these models have in common that they are differentiable end-to-end versions of models inspired by classical computability. This includes the popular *Long Short-Term Memory* (*LSTM*) architecture [204], or the so-called *Differentiable Neural Computers* [188], or the *Neural Turing machines* [187]. The underlying principle of these constructions is to extend an artificial neural network by coupling it to an external memory resource. This external resource can also be a stack as in the so-called *Neural Network Pushdown Automata* [318] and *Neural Stack machines* [189].

The models discussed previously all work in discrete time, but models with a continuous-time dynamics have also been considered, such as *symmetric Hopfield networks*. The convergence behavior of these networks has been used in various applications such as associative memory, or combinatorial optimization problems after first applications in [210].

Models based on *spiking neurons* have been claimed to be biologically more realistic [233]. Various coding methods exist to interpret the outgoing spike train as a real-number value either relying on the frequency of spikes, or the timing between spikes. This yields various ways of encoding information and the computational power of spiking neuron models has been investigated in a series of papers: see [309], [233] or [177] for surveys. Recent years have seen practical implementations [246] as well as their use to solve practical problems efficiently [148].

6.2.4 Models from Verification

The development of algorithms and techniques for verification or control of so-called *hybrid systems* or *cyberphysical systems* have also generated several lines of research related to analog computation. *Hybrid* and *cyberphysical systems* have in common that they mix discrete evolutions, often a digitally engineered controller, with continuous dynamics that often comes from the environment or from some natural continuous variables that the controller acts upon.

There is an extensive literature on the *hybrid automata* modeling approach to determine the exact frontier between decidability and non-decidability for reachability properties, according to the type of dynamics, guards and resets allowed. The same holds for the frontier about applicability of techniques coming from control to prove properties about these systems such as stability. Providing a complete panorama on this literature is out of the scope of the current chapter. Classical survey references are [202, 14]. See also [64] for discussions.

This literature is the source of many *dynamic undecidability* results, to which we will come back in Section 6.3.3.

Hybrid systems are known to exhibit the so-called *Zeno phenomenon*. In short, they may happen to have an unbounded number of discrete transitions in a bounded (continuous-) time. In this context, it is classical to distinguish various types of Zeno behavior: *Chattering Zeno* versus *Genuine Zeno*, following [16]. The first type can often be eliminated and corresponds more to an artifact of the model, which can be avoided by considering an appropriate notion of solution, while the second is more problematic to detect and harder to avoid in simulation [17]. We will come back in Section 6.4.4 to Zeno's phenomenon in the wider perspective of spacetime contraction phenomena in analog systems.

Despite some promising early results in the field, open problems even for very simple classes of systems still remain. In particular, the decidability of the reachability problem for *piecewise affine maps on the real line* is a famous open problem, with some partial progress such as [221, 46]. Several other questions seem to be closely related to that issue [31]. The decidability of *point to hyperplane* reachability for discrete-time linear systems, known as the Skolem problem, is famously open. The *continuous-time* version of *point to hyperplane* has recently been shown to be related to conjectures from transcendental number theory [121]. Hardness of *recurrent reachability for continuous linear dynamical systems* of low dimensions has been investigated in [122].

Several recent results have also focused on the hardness of *bounded-time versions* of reachability problems. The complexity of the reachability problem has been characterized for hybrid automata in [86], and for piecewise affine systems in [47] and [38]. The complexity of problems or methods from control theory has also been explicitly derived [11, 9, 10].

In relation to other chapters of this handbook, we mention that the recursive analysis approach has also been explored. Computability of reachable and invariant sets in the framework of computable analysis have been investigated for continuous-time systems [125] and for hybrid systems [126].

A method to approximate hybrid systems with a polynomial hybrid automaton, i.e., a Taylor approximation, has also been proposed [223], as well as interval methods and Taylor model methods for ODEs [264]. There have been attempts to provide a formal semantics to Simulink based on non-standard analysis tools [52]. More generally, providing models of systems using ordinary differential equations is a hard task in practice: see e.g. very instructive discussions in [224] about all the difficulties in modeling a simple system such as Newton's pendulum.

6.2.4.1 Timed Automata

Timed automata [13] can be considered as a restricted version of hybrid systems for which decidability of reachability holds. They can also be considered as an extension of finite automata with clocks. The model has clear practical applications and is at the heart of several computer tools for verification: see e.g. recent survey [77].

From a more fundamental point of view, timed automata can be seen as language recognizers [13], and there have been various attempts to generalize concepts from finite automata theory to this framework. This includes closure properties of recognized languages [13], pumping lemmas [40] as well as variants of Kleene's theorem [25, 26, 27, 78, 79, 28]. For a recent survey about timed automata, see [77].

6.2.5 Blum-Shub-Smale's Model

The *Blum-Shub-Smale (BSS) model* [57] has been introduced as a discrete-time model of computation over the reals in order to discuss hardness of problems in algebraic complexity. In the initial presentation of the model, operations of the field $\mathbb{R}$ are assumed to be realizable at unit cost, leading to classes such as $P_{\mathbb{R}}$ and $NP_{\mathbb{R}}$, with complete problems such as the existence of a real zero of a given polynomial. Later, it has been generalized to other fields or rings with extended or restricted operations [56], or to an abstract model over arbitrary logical structures [283]. Notice that classical discrete computability models have been generalized to abstract structures in various ways in parallel, and also before [57]: see e.g. [172], [258] or [323].

The obtained computability and complexity theory subsumes classical discrete computation theory since the latter can be seen as the specific case of logical structures with a finite domain.

There is an extensive literature on the related computation theory. Many results have been obtained in this model, mostly studying the corresponding complexity classes and their relations with classical questions in computability theory, or providing lower or upper complexity bounds on various problems based in this framework: see [56, 283].

Recent results include non-trivial generalizations to this framework of Toda's theorem [35, 36] or the PCP theorem [34], interactive proofs [33], Ladner's result [243] as well as separation of degrees [176, 175].

The model is different in spirit from most others discussed in this section, as it is usually not considered to be an attempt at modeling analog machines in a realistic way, but rather as a mathematical tool. It has proven to be relevant for the discussion of lower bounds in algebraic complexity, or for some classical questions from complexity theory in a wider generalized setting, where sometimes complexity classes such as P and NP can be separated. The model also has clear connections with the generalized finite automata models discussed in Section 6.5.3.

6.2.6 Natural Computing

The interest in unconventional models of computation, and in particular in natural computing, has revived interest in analog computing.

6.2.6.1 Dissipative Influence Dynamics

The framework of *natural algorithms* has motivated a series of works about models of *influence systems* and their computational capabilities. A manifesto in favor of the fact that the study of natural systems can benefit from an algorithmic perspective has been published in [114]. This has motivated the exploration of several models of influence dynamics such as the Hegselmann-Krause's model. In particular, bounds on the time required by a group of birds to stabilize in a standard bird-flocking model have been established in [115]. Turing completeness and almost sure asymptotic periodicity of diffusive influence systems have also been obtained in [117]. For a general discussion on the merits and challenges of an algorithmic approach to natural algorithms, see [116].

6.2.6.2 Physarum Computing

Physarum polycephalum is a slime mold that has been shown to be able to solve various natural problems: this includes realization of Boolean logic gates, implementation of delay in computing circuits, geometry computations such as Voronoi diagrams or Delaunay triangulations, and computation of shortest paths [6]. Models of various aspects of its behavior have been established. Convergence proofs and complexity bounds for computing shortest paths have been investigated [41], based on a mathematical model for the slime's behavior proposed in the form of a coupled system of differential equations [320]. Implementation of Kolmogorov-Uspensky machines in a biological substrate has also been investigated [4]. For a survey about these fields of research, see [6].

6.2.6.3 DNA and Molecular Computing

Areas such as *systems biology* aim at understanding complex biological processes in terms of their basic mechanisms at the molecular level. Many attempts to apply concepts and tools from theoretical computer science to this framework (logic, algebra, etc.) have been investigated, with a lot of success and concrete software systems: see [163].

However, even if the primary purpose of these fields was to explain concrete biological features, various approaches have considered computations by chemical reactions as a programming tool. The idea is to consider computations by reactions as programs to solve various tasks, as nature does in the context of cell biology, but not restricted to this context. In particular, various attempts to relate this concept of programming to computation theory have been proposed. In most of these approaches, the underlying principle is to get inspiration from concrete chemical phenomena in order to derive abstract models of computation which are potentially usable. This includes the model of biomolecular computation [196], the *Chemical Abstract Machine* [53], *Membrane Computing* models [271], *Biochemical Ground Form* process algebra approaches [108], DNA-based computation models [285], and *Chemical Reaction Networks* (CRN), discussed in Section 6.6.5.

Simulation of Turing machines has been demonstrated in several of the abovementioned articles, either at an abstract level or concretely. A very challenging question is to compare the actual implementations in nature of some of the tasks to other possibly more efficient ones that could be derived theoretically [164].

For a presentation of natural computing models and results not covered in this chapter, see [291, 270]. In this chapter, we will only briefly discuss the case of DNA computing (below) as well as that of CRNs (Section 6.6.5).

DNA computing started with a work [8] which proposed to solve the directed Hamiltonian path problem on a graph using DNA as well as enzymes to implement the computation. Molecular computation had been investigated in various ways before Adleman's popular article (see e.g. [128]), including the idea of using DNA computing to implement computations of formal language theory [199]. This was later extended to other known NP-complete problems [228], even if the required resources have been demonstrated to be unrealistic. Approaches to control resources based on evolutionary computation have also been proposed [317]. The field developed meanwhile in many impressive ways. Visible facts include the development of a programmable molecular computing machine [232], the demonstration of the use of DNA as a digital storage medium encoding a 5.27-megabit book [123], or the construction of the analog of a DNA transistor [60].

6.2.7 Solving Various Problems Using Dynamical Systems

Several authors have shown that certain, possibly discrete, decision or optimization problems such as graph connectivity or linear programming can be solved by spe-

cific continuous dynamical systems. Some examples and references can be found in the papers [310, 327, 89, 166, 200, 49, 257, 55]. This is linked to various mechanical computer models (e.g. "billard ball computers") investigated by several papers in the 1970s and 1980s, with discussions about physical limits on computations such as thermodynamic reversibility [50, 171, 51, 241]. Relevant concepts have turned out to be still important in quantum computing (e.g., Toffoli gates, etc.). This is also related to discussions about various phenomena possibly capable of hypercomputation [242, 298]: see in particular Section 6.2.9 about Black Hole computations for more recent references.

Observe that, if analog computation is to be understood in the sense of computing by analogy, then almost any historical analog machine can be considered as falling under this framework. In his monograph, Ulmann develops sometimes with great and instructive details some of the machines for particular problems such as finding zeros of a polynomial, linear algebra, optimization and simulation [326]. In more than 70 pages various historical applications in about 20 fields such as Mathematics, Physics, Mechanics, Geology and Economics are described.

The question of whether some of the discrete problems could actually be solved faster using continuous methods is very intriguing: this is the object of Section 6.4.4.

Many dynamical systems of the form $H' = [H,[H,N]]$, where the notation $[B,L]$ stands for $BL - LB$, have been shown to be continuously solving some particular discrete problems [89, 88] such as sorting lists, diagonalizing matrices and linear programming problems. One key property is that this equation is equivalent to some gradient flow in the space of orthogonal matrices. Many examples in this spirit are discussed in detail in [201]. More recently a system of nonlinear differential equations of a similar form to sort numbers fed to the input has been investigated [181].

Notice that several discrete-time algorithms or methods have some analog equivalent. This includes *Newton's method*, which leads to *Newton's flow dynamics* for finding roots. *Gradient descent* methods (this includes the very popular backpropagation method for Neural Networks) can also be considered as the (explicit) Euler's discretization method of a continuous flow, the so-called *gradient flow*.

The use of analog methods for solving k-SAT problems has been investigated in [159]: the problem is mapped to an ordinary differential equation about which some properties are established, such as chaotic transience of trajectories above a constraint density threshold and fractality of the boundaries between the basin of attraction. The system is stated to always find solutions in polynomial continuous-time, but at the expense of exponential fluctuations in its energy function [159]. An attempt to physically implement this dynamic has been proposed [335]. Previous statements that k-SAT can be formulated continuously can be found in [191, 263, 328].

Notice that polynomial-size continuous-time Hopfield nets have been proven able to simulate PSPACE Turing machines [308]. This implies that even ODEs with Lyapunov function-controlled dynamics can actually do much more than solving NP-complete problems in some sense.

6.2.7.1 Interior Point Methods

A particular class of methods falls very naturally into this framework: the so-called *interior point methods*, which correspond to a particular class of algorithms for solving linear and nonlinear convex optimization problems. The principle, already proposed by John von Neumann and very popular in the 1960s, is to build a continuous system whose trajectories evolve in the interior of the feasible region. A common method to guarantee evolution in the feasible regions was the use of *barrier functions* acting as a potential energy. By making these functions tend to infinity on the boundary, the evolution is guaranteed to remain feasible.

Later, these methods were mostly considered to be inefficient until Karmakar triggered a revolution in the field of optimization by providing the second polynomial-time algorithm for linear programming [215, 213]. It was then realized that Karmakar's algorithm is equivalent to a particular interior point method [178, 37]. Notice that, historically, the first polynomial-time algorithm for linear programming is due to Khachiyan and is based on the ellipsoid method (which is not an interior point method).

A very elegant presentation of Karmakar's algorithm and the associated flow, inspired from [23], can be be found in [257], including an elegant presentation of its polynomiality based on ordinary differential equation arguments. Refer to [288] for a general introduction to interior point methods.

6.2.8 Distributed Computing

Recent years have also seen the birth of new classes of models in *distributed computing*. In particular, the model of *population protocols* consists of passively mobile anonymous agents, with finitely many states, that interact in pairs according to some rules, i.e., a given program [20]. *Passively mobile* means that agents have no control over the other agents with whom they will interact. The model was initially introduced in the context of sensor networks, but it is nowadays considered to be a fundamental model for large passively mobile populations of agents with resource-limited anonymous mobile agents.

Most works on the model have considered these protocols as computing predicates over multisets of states. Given some input configuration, the agents have to decide whether this input satisfies a given predicate: the population of agents has to eventually *stabilize* to a configuration in which every agent is in a particular accepting (respectively rejecting) state if and only if the predicate is true (resp. false). The model is *uniform*: the program is assumed to be independent of the size of the population.

The seminal work of Angluin *et al.* [21, 20] proved that predicates decided by population protocols are precisely those on counts of agents definable by a first-order formula in Presburger arithmetic – equivalently, this corresponds to semilinear sets. An elegant proof of this result can be found in [160]. Note that this computa-

tional power is rather restricted, as multiplication for example is not expressible in Presburger arithmetic.

The model has been intensively investigated since its introduction. Several variants have been studied in order to strengthen it with additional realistic and implementable assumptions. This includes natural restrictions like modifying hypotheses on interactions between agents (e.g., one-way communications [21], particular interaction graphs [19]). This also includes probabilistic population protocols that assume random interactions [20]. Fault tolerance has also been considered [144], including self-stabilizing solutions [22]. For some introductory texts about population protocols, see for instance [32, 248].

The model can be seen as a particular case of the (stochastic) chemical reaction networks discussed in Section 6.6.5.

Here, we focus on models and results in the context of distributed computing, and mainly discuss computability issues. Covering all works devoted to variants of population protocols in the distributed computing community is out of the scope of this chapter: see [32, 248] for surveys. Among many variants of population protocols, the *passively mobile (logarithmic-space) machine model* was introduced by Chatzigiannakis *et al.* [113]. In this model, each agent carries a bounded-space Turing machine, instead of a finite state automaton. In an orthogonal way, *community protocols*, where each agent has a unique identifier and can only store $O(1)$ other agent identifiers, exclusively from agents that it has met, were introduced by Guerraoui and Ruppert [192]. They proved, using results about the so-called *storage modification machines* [302], that such protocols simulate Turing machines very efficiently. A hierarchy between the two models has also been studied recently, by considering the case of homonyms, that is to say when several agents may share the same identifier [66, 67].

The population protocols can also capture natural models of dynamics of some opinion-spreading models by considering probabilistic rules of interactions: results on the convergence and threshold properties of the so-called Lotka-Volterra population protocol have been established [141].

Notice that many models coming from dynamics of rational agents in the context of (learning equilibria in) Game Theory can also be considered as analog distributed models of computation [329, 205].

6.2.8.1 Large-Population Protocols

When considering probabilistic interaction rules, as in the previously mentioned settings, the underlying dynamical system is a Markov chain.

If the population of agents is large, the random process converges to its (deterministic) limit continuous dynamic given by some ordinary differential equation (also called its *mean-field limit*): this corresponds to the differential semantics discussed in Section 6.4.1 for chemical reactions.

These considerations led to the so-called *Large-Population Protocols* [70]: real numbers which correspond to limit ratios of programmable dynamics by such mod-

els have been demonstrated to correspond precisely to algebraic numbers [70]. This result has many similarities with [211], obtained in the context of stochastic reaction networks, but with a slightly different notion of computability.

A framework to translate certain subclasses of differential equations into protocols for distributed systems has been proposed [193]. This is illustrated on several examples either taken from distributed problematics (responsibility migration, majority selection) or from classical models of populations such as the Lotka-Volterra model of competition.

6.2.9 Black Hole Models

Black Hole computation usually refers to the study of the validity of the Physical Church Thesis (see Section 6.4.2) in the context of Einstein's General Relativity (GR). In other words, does GR allow for spacetimes where an observer can observe in finite time an eternity of some other device. Informally, the setup is as follows: some device (that we refer to as the computer) will try to solve some hard problem, like checking the consistency of ZFC. As soon as the computer finds a counterexample, it sends a signal to the observer, otherwise it keeps checking ZFC for eternity (since there is an infinite number of formulas to check). In parallel, an observer will manage to view the result of this infinite computation in finite time. More precisely, the observer will have a spacetime location q, at which it can check for the existence of the signal: if it receives a signal, then ZFC is inconsistent, otherwise ZFC is consistent. Hogarth proved that such spacetimes, usually referred to as *Malament-Hogarth* (*MH*), can exist in theory [209]. Note that in this setting, it is crucial that q is known to the observer, so that after it has reached q, the observer knows whether ZFC is consistent or not, and can use this information.

This question has received a lot of attention, especially concerning the physical realization of such a setup [158, 206, 207, 161, 208]. It has since emerged that *slowly rotating Kerr black holes*, and possibly *Reissner-Nordström* (*RN*) black holes, provide a plausible physical realization for these experiments. Two very good surveys on the problems and solutions to the many obstacles encountered so far have been published [265, 266]. We will mention some of these in the remainder of this section. Another important question is to characterize the extent of hypercomputation available in an MH spacetime. Hogarth originally showed that any Σ_1 set could be decided in the Kerr spacetimes. Recently, it was shown that MH (but not necessarily Kerr) spacetimes can decide all hyperarithmetic predicates on integers, but not more (under some assumptions) [331].

Without giving too many details, we now mention some aspects of the physical realization of black hole computation. An historical objection was that a *Schwarzschild black hole* (non-rotating and non-charged) has a punctual singularity to which any observer is attracted (in this setting, the observer is the one "jumping" into the black hole whereas the computer stays outside), and eventually gets crushed by the ever-growing tidal forces that tear it apart. However, both slowly rotating Kerr

and RN black holes have negligible tidal forces. Furthermore, their singularity has a shape of ring that the observer can avoid forever, meaning that the observer could survive infinitely long within the black hole. In particular, astronomical evidence suggests that such Kerr black holes exist and have a size roughly that of a solar system. Another kind of objection was that this setup neglected quantum effects that do not exist in pure GR, but many of these objections have been resolved, although the details depend on whether the universe is expanding or not. To summarize with a slightly exaggerated statement, the entire Earth population could jump into a black hole, and continue living inside forever but now knowing whether ZFC is consistent or not. Other spacetimes, such as the *anti-de Sitter*, theoretically allow the roles of the observer and the computer to be exchanged, meaning that the observer could stay out of the black hole and send the computer into the black hole, and get an answer in finite time.

Finally, we mention some related work on *Closed Timelike Curves* (*CTC*s) that, if they exist, would allow PSPACE problems to be solved very quickly but do not allow for hypercomputation [3]. Another line of research is to build a logical axiomatization of spacetime, that is, to build relativity theory as a theory in the sense of first-order logic. For a survey on the subject, see [18].

6.2.10 Spatial Models

We now consider various *spatial* models, that is to say various models which share a concept of computation based on the distribution of entities in some space and communications between these entities. We agree that our classification is debatable: some of the previous models can already be considered as such (see e.g. reaction-diffusion systems), and some of the considered models can also be considered as continuous space (see e.g., cellular automata that can be considered as acting over Baire's space $\{0,1\}^{\mathbb{N}}$ homeomorphic to $\mathbb{R}$).

6.2.10.1 Computational Fields

Computational fields are analog models of spatial massive parallelism. They are motivated by the hypothesis that in various contexts it now makes sense to consider that the number of processing elements is so large that it may conveniently be considered as a continuous quantity [238, 236]. The theoretical functions of computational fields, based on tools from functional analysis, have been studied [236]. Applications of computational fields have been explored and advocated in several articles [240, 238, 237].

6.2.10.2 Spatial Computing Languages

Previous computational field models fall naturally into the more general framework of *spatial computing*: computation has become cheap enough and powerful enough that a large number of computing devices can now be embedded into many environments. As a result, a whole spectrum of *Domain Specific Languages* (DSLs) have emerged to narrow the gap between the application needs of users in various domains (e.g., biology, reconfigurable computing), usually at a global level, and the programming, usually at a local level, of the increasingly complex systems of interacting computing devices. A common pattern in all those models and languages is the close relationship between the computation and the arrangement of the computing devices in space. For a survey and references, with discussions about many related aspects, see [39].

6.2.10.3 Cellular Automata-Based Models

Cellular automata correspond to a particular class of spatial computing models. The classical model is an abstract discrete-time and -space model, with a rather well-studied computation theory. We will focus here on developments related to analog models.

There exist several Cellular Automata (CA)-inspired models, usually built as a abstraction of CA in the limit where cells are infinitely small.

One such model is based on an infinite tessellation of spacetime [301]. It has been compared to classical models from computability theory and proven to be capable of hypercomputations [301].

Smooth (continuous) versions of the Game of Life have been investigated [203], yielding very elegant dynamics. More generally, the passage from cellular automata to continuous dynamics (partial differential equations) has also been investigated in [111].

Another model with several recent developments is that of *Signal Machines* (SM) introduced in [152] where dimensionless signals are synchronously moving on a continuous space, and local update rules are used to resolve collisions. The major difference with CA is that both time and space are continuous (i.e., reals instead of integers). A SM configuration is given by a partial mapping from $\mathbb{R}$ (the space) to a finite set of *meta-signals*. Each SM comes with a finite set of *collision rules*. When two or more signals meet, a *collision* happens: all incoming signals are destroyed and the rule gives a list of outgoing signals that are created. Between collisions, each signal moves in a direction at a certain speed that depends on its type. Thus there are only finitely many different speeds. One can think of the speed of a signal as the *slope* of its line in the 2D space-time diagram. Figure 6.3, illustrates this process on a simple example. In a series of papers ([153] and onwards), Durand-Lose and coauthors investigate the computation power of this model with various restrictions. It is possible to embed Black Hole computations, Turing machines, BSS machines and more into this model, depending on whether irrational numbers and

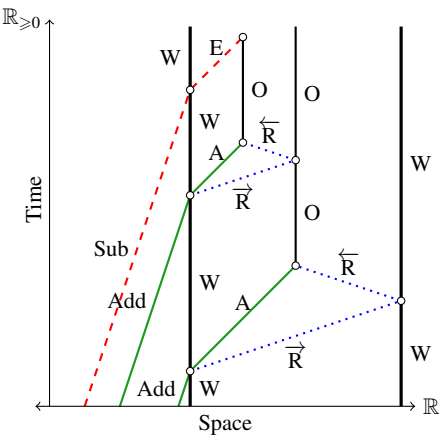
accumulation points are allowed. Since the unrestricted model exhibits the Zeno phenomenon, it is super-Turing powerful.

Name	Speed
Add, Sub	1/3
A, E	1
O, W	0
$\overrightarrow{R}$	3
$\overleftarrow{R}$	-3

(a) meta-signals

$$\{ \text{Add}, \text{W} \} \to \{ \text{W}, \text{A}, \overrightarrow{R} \}$$
$$\{ \overrightarrow{R}, \text{W} \} \to \{ \overleftarrow{R}, \text{W} \}$$
$$\{ \text{A}, \overleftarrow{R} \} \to \{ \text{O} \}$$
$$\{ \overrightarrow{R}, \text{O} \} \to \{ \overleftarrow{R}, \text{O} \}$$
$$\{ \text{Sub}, \text{W} \} \to \{ \text{W}, \text{E} \}$$
$$\{ \text{E}, \text{O} \} \to \{ \}$$

(b) collision rules



(c) space-time diagram

Fig. 6.3 A signal machine that finds the middle between two input vertical signals. The process is started by the arrival of the Add signal from the left. It triggers a sequence of collisions that results in a vertical signal O that is positioned exactly halfway between the two W signals. It is possible to generate the middle between the left W and O signal by sending another Add signal. It is also possible to suppress the leftmost O signal by sending a Sub signal from the left. Finding the middle is a key primitive in SM computations.

6.2.11 Various Other Models

In this chapter we do not intend to cover all models. Various other unconventional computers can also be classified as spatial. This includes models based on propagation and interaction of waves (reaction-diffusion computers [300, 299, 7], soliton-based computers), models based on propagation of pattern forms (slime mold computers [5], plant root computers, crystallization computers), and models based on propagation of swarms of creatures (e.g., crab gates). This also includes molecular self-assembly models [162, 316].

The frontier between analog and non-analog models of computation is not so clear, and some models are clearly at the frontier. This includes the optical models of [334], or some quantum computation models such as [147, 190, 304, 216], and finite two-dimensional coupled map lattices, which have been proven to be computationally universal [268].

Planar mechanisms were claimed to correspond to all finite algebraic curves by Kempe [214]. The initial statements of Kempe had flaws that were corrected and ex-

tended in [310]. *Tractional motion machines* constitute an extension of this model with a rich and elegant theory discussed in [250, 251]. The fact that some functions computable in this model are not GPAC computable has been established [250]. Some discussions on limitations of machines based on the historic concept of compass and ruler constructions can be found in [259].

6.3 Dynamical Systems and Computations

6.3.1 Arbitrary Versus Rational/Computable Reals

Before delving into the core of the topic of *dynamic undecidability*, we would like to point out a fundamental difference between discrete and continuous dynamical systems: *real numbers*. Indeed, it is clear that many objects involved in discrete computations (integers, rational numbers, graphs, etc) have a finite representation. For example, one can represent an integer in base 2 by its bits. This property is no longer true for real numbers, independently of the details of the representation. In fact, for reasonable representations (see [330] for more on the theory of representations) of the real numbers, a *single real number* can turn out to be a very powerful object. One such example is the real number, sometimes called the *"Turing number"*, whose n^{th} bit is 1 if and only if the n^{th} Turing machine halts[1]. Intuitively, any powerful enough dynamical system that has access to this number can solve the Halting problem for Turing machines in finite time, and is thus super-Turing powerful. Other numbers, such that Ω of Chaitin [112], can turn out to be even more powerful. A consequence of this fact is that dynamical systems are able to solve (classically considered) uncomputable problems if one allows such numbers to appear in the system. This is why it is often important to restrict restrict our attention to systems with computable or rational descriptions if one does not want to consider models with some hypercomputational features.

Understanding the obtained computability theory, if this hypothesis is not assumed, i.e., when arbitrary (possibly non-computable) reals are allowed in the description of systems, gave birth to a whole set of developments discussed in Sections 6.6.1 and 6.6.2.

In order to avoid the consideration of such classes of models, we assume for the rest of this section that we restrict ourselves to rational or computable descriptions.

6.3.2 Static Undecidability

It is, however, often rather easy to get undecidability results about dynamical systems by observing that integers (or discrete sets) can be considered as particular

[1] One needs to fix a particular enumeration of Turing machines.

reals, and hence that all undecidability problems known over the integers have some counterpart over the continuum [296]. This allows us to map well-known undecidability questions of recursive analysis (e.g., testing whether a real is zero) or of computability theory (e.g., testing whether a given polynomial with integer coefficients has an integer root) into the setting of dynamical systems. This is sometimes known as *static undecidability*. For concrete simple examples of results obtained in this vein, see [24, 296]. More elaborate examples are the proof that determining whether a polynomial dynamical system has a Hopf bifurcation [137] or is chaotic [138] is undecidable.

6.3.3 Dynamic Undecidability

Dynamic undecidability, as opposed to *static undecidability* [296], corresponds to a proof technique that is often used with various models of dynamical systems. In this case, the undecidability is obtained by proving that for every Turing machine, one can build a dynamical system in the considered class that is able to simulate the Turing machine step by step.

To be more concrete, let us informally describe an embedding of Turing machines into dynamical systems with discrete time and continuous space. We can encode the configuration of a Turing machine using a vector (l,q,r) where $l,r \in [0,1]$ will encode the tape (left and right side of the tape), and $q \in [0,1]$ will encode the state, for example the i^{th} state is mapped to $\frac{i}{N}$ where N is the number of states. To encode the tape, we simply view the content of the tape as a list of digits. In the case of a binary alphabet, a word $w \in \{0,1\}^*$ will be encoded by the real $0.w = \sum_{i=1}^{|w|} w_i 2^{-i}$. By convention, we will say that the symbol under the head is the first symbol on the right tape.

To simulate one step of the machine, we need to perform tests on the state and then apply the corresponding action. To find in which state we are, we need comparisons of the form $q =^? \frac{i}{N}$. To find which symbol is under the head (the first symbol of the right tape r), note that if $r_1 = 0$ then $0.r < \frac{1}{2}$, but if $r_1 = 1$ then $0.r \geqslant \frac{1}{2}$. Thus we only need a test of the form $r < \frac{1}{2}$. To move the head of the machine, we will only need addition and multiplication by a constant. For example, if we want to move the tape to the right, then we perform the assignment $r \leftarrow \frac{\sigma+r}{2}$ where σ is the symbol written by the head, and $l \leftarrow 2(l-\rho)$ where ρ is the first symbol of the left tape. Note that we can easily have access to ρ the same way we have access to σ and we can make a case distinction between $\rho = 0$ and $\rho = 1$. Finally we assign the new state with $q \leftarrow \frac{i'}{N}$ where i' is the new state.

To summarize, in order to simulate a Turing machine with a discrete-time and continuous-space machine, we need comparisons against fixed values ($= x$ and $< x$), assignment, addition and multiplication by a constant. Importantly, all the constants involved are rational numbers. One may be worried that exact comparisons between real values is an unrealistic assumption; there are two answers to this issue. First,

analog models of computation are not necessarily concerned with realistic assumptions that come from the world of discrete computations (or from computable analysis's basic assumptions about computability): comparing two real numbers is a very natural operation in several continuous models of computation.

Second, and more importantly, even if exact tests are not possible, this simulation does not really require exact tests. Without going into too many details, notice that on the state, there is gap of size $\frac{1}{N}$ between the encoding of two states, thus an approximate test will suffice. Similarly for the tape, a classical trick is to encode a binary tape in base 4, by $0.w = \sum_{i=1}^{|w|} 2w_i 4^{-i}$. This way, if $r_1 = 0$ then $0.w \leqslant \frac{1}{6}$ (the worst case is $0.02222\ldots$) but if $r_1 = 1$ then $r \geqslant \frac{1}{2}$, leaving again a constant gap between the two options.

All the above settings provide the basic ingredients for the possibility to simulate any Turing machine by a discrete-time model. Similar *static undecidability* constructions have been used in various models: general dynamical systems [255, 294, 82, 296, 127], piecewise affine maps [218], sigmoidal neural nets [307], closed-form analytic maps [219], which can be extended to be robust [185], and one-dimensional restricted piecewise-defined maps [221].

If one wants to simulate a Turing machine by a continuous-time model, then several techniques may be used. One is to build a continuous-time system such that the crossing of the dynamics with a given hyperplane (its *Poincaré map*) corresponds to a discrete-time system as above. Another one is to build a continuous-time system such that its projections at discrete time $t = 0, 1, \ldots$ (its *stroboscopic view*) is given by a discrete-time system as above.

Usually such constructions rely on a suitable generalization of the idea of *"continuous clocks"* [82] for iterating functions over the integers. For simplicity, we will use a reformulation of these original equations, based on [101]. The idea is to start from the function $f : \mathbb{R} \to \mathbb{R}$, preserving the integers, and build the ordinary differential equation over $\mathbb{R}^3$

$$\begin{aligned} y_1' &= c(f(r(y_2)) - y_1)^3 \theta(\sin(2\pi y_3)) \\ y_2' &= c(r(y_1) - y_2)^3 \theta(-\sin(2\pi y_3)) \\ y_3' &= 1, \end{aligned}$$

where $r(x)$ is a rounding-like function that has value n whenever $x \in [n - 1/4, n + 1/4]$ for some integer n, $\theta(x)$ is 0 for $x \leq 0$ and $\exp(-1/x)$ for $x > 0$, and c is some suitable constant. A simple analysis of this dynamics shows that it basically alternates $y_1 \leftarrow f(y_2)$ and $y_2 \leftarrow y_1$. As a result, the stroboscopic view of this differential equation is the discrete-time system $y(n+1) = f(y(n))$.

Indeed *dynamic undecidability* constructions have been used in various continuous-time models, including piecewise constant maps [30], general dynamical systems [82] and polynomial ordinary differential equations [185].

Observe that dynamic undecidability has merits not covered by static undecidability. Considering Turing machines as dynamical systems provides a view not covered by the von Neumann picture [101]. This also shows that many qualitative features of (analog or non-analog) dynamical systems, e.g., questions about basins

of attraction, chaotic behavior or even periodicity, are not computable [255] even for a fixed dynamical system (corresponding for example to a universal Turing machine). Conversely, general questions traditionally related to dynamical systems are brought by this into the realm of Turing machines and computability [64]. These include, in particular, the relations between universality and chaos [24], necessary conditions for universality [145], computability of entropy [217] and understanding of the edge of chaos [225].

One may object that the above simulations require unrealistic hypotheses on systems. For example, that they require the ability to do computations with arbitrary precision: $O(2^{-n})$ precision corresponds roughly to size $O(n)$ of the tape in the simulation above. But at this point, in order not to overinterpret such statements, we believe that some digression about computational models would be clarifying. This is the topic of the next section.

6.4 Philosophical, Mathematical and Physics-Related Aspects

6.4.1 Mathematical Models Versus Systems

While the use of dynamical systems is very common in the experimental sciences (biology, physics, chemistry, etc.), it is important to remember that this is an abstract (mathematical) view, and that the properties of those systems can actually be different in many aspects from those of the initial system they intend to model.

In the literature, there are many accounts in various contexts about situations where models behave very differently according to the levels of modelization. As an illustrative example, we can mention the Lotka-Volterra equations discussed in [226, 220]. It is shown in these articles that no classical ordinary differential equation method (unless biased on purpose) behaves as stated in classical studies of these equations: no closed cyclic behavior is observed, for example, and "energy"-like functions supposed to be preserved are not. While the continuous Lotka-Volterra dynamic is usually presented as an abstraction of a discrete setting, these results can be seen as a non-matching of behaviors between discrete and continuous models.

A context where various semantics have been discussed, with clearly stated vocabulary, is that of chemical reactions. To a given formal set of chemical reactions, a hierarchy of semantics can be associated at different levels of abstraction [163].

The most concrete (low-level) interpretation is provided by the *Chemical Master Equation (CME)*, which defines the probability of being in a state x at time t by considering the system as a continuous-time Markov chain governed by *propensity* of reactions over a discrete number of molecules, giving the rate of reactions of the associated chain. The *differential semantics* describes the evolution of the system by an ordinary differential equation (ODE). This ODE can be seen as the mean-field view of the reactions, and can be considered as being derived from the CME by a first-order approximation. Then the *stochastic semantics* is defined by considering a

Continuous-Time Markov Chain (CTMC) over integer numbers of molecules (discrete concentration levels). The very classical algorithm of Gillespie [180] provides a method to efficiently simulate this CTMC from the reactions. If the simulation provided by this algorithm is often similar to the ODE simulation for a large number of molecules, it may exhibit qualitatively different behaviors, in particular when the number of molecules is small [163].

The abstraction of stochastic semantics by simply forgetting the probabilities also yields the *Petri net semantics* of the reactions, where the discrete states define the number of tokens in each place, and the transitions consume the reactant tokens and produce the product tokens. The abstraction of the Petri net semantics into *Boolean semantics* is then obtained by reasoning only on the absence/presence of a given molecule.

All these discrete and stochastic trace semantics of reaction systems can be related by a Galois connection in the framework of abstract interpretations [165]. However, it is important to always remember that the behavior of a given semantics can differ from the others in many ways.

The study of the relationship between the differential and the stochastic semantics dates back to the seminal work of Boltzmann in the nineteenth century, who created the domain of Statistical Physics. In this setting, the differential semantics is obtained from the stochastic semantics by limit operations, where the number of molecules tends to infinity and the time steps to zero, under several assumptions such as perfect diffusion [165]. More generally, there is a whole mathematical theory justifying the passage from a stochastic dynamic over a (usually huge) population of agents to its first-order ordinary differential equation description [222]. However, hypotheses of mathematical theorems are not always valid in the context of experimental science applications and concerns are sometimes very different. Notice that the specific context of chemistry has been discussed in articles such as [179].

All these discussions emphasize the difference between a system and its models, and even between the various levels of abstraction of a given system, and the importance of stating explicitly the considered semantics in the coming discussions.

6.4.2 Church-Turing Thesis and Variants

A common statement of the Church-Turing Thesis is that *every effective computation can be carried out by a Turing machine, and vice versa.* However, it is often misunderstood, and an instructive discussion demonstrating the improper semantic shift from its initial statement and the way it is often misunderstood today can be found in [133]. We repeat some of these elements here.

Following [133], "The Church-Turing thesis concerns the concept of an effective or systematic or mechanical method in logic, mathematics and computer science. "Effective" and its synonyms "systematic" and "mechanical" are terms of art in these disciplines: they do not carry their everyday meaning."

A myth seems to have arisen progressively in several documents about the fact that Alan Turing's 1936 paper was establishing facts about limitations of mechanisms [133]. The thesis that *whatever can be calculated by a machine* (working on a finite data in accordance with a finite program of instructions) *is Turing machine-computable* (sometimes referred to as the *Physical Church-Turing thesis*, or *Thesis M* of [174]) is a very different statement from the Church-Turing thesis.

Thesis M itself admits two interpretations, according to whether the phrase *can be generated by a machine* is taken in the sense *can be generated by a machine that conforms to the physical laws (without resource constraints) of the actual world*, or in a wide sense whether the considered machine could exist in the actual world [133]. Under the latter interpretation, the thesis is clearly wrong, and hence not really interesting: see all examples of hypercomputations mentioned in the next subsection.

The discussion above suggests that it may also be important to distinguish machines from their models. The thesis that *whatever can be calculated by a given model of machines in model(s) of our physical world is Turing machine-computable* might still be one more different statement.

6.4.3 Are Analog Systems Capable of Hypercomputations?

Several results have shown that analog systems are capable of hypercomputations. One classical way to do so is to consider non-computable reals in the machine (see the previous discussion in Section 6.3.1 and coming Sections 6.6.1 and 6.6.2). Another classical way is to use Zeno's phenomenon: the possibility of simulating an unbounded number of discrete steps in bounded continuous-time (e.g., [29, 256, 61], see Section 6.4.4). A presentation of various systems capable of hypercomputations, with in particular some accounts on analog models of computations, can be found in [319].

Such results may give some clues about the fact that the considered model or systems are indeed very/super powerful. But, as already stated in [239], we believe that the main interest of such results is not really to advocate models with hypercomputational power, but rather a way to demonstrate that an extended definition of computation and computability theory including alternative (especially analog) models in addition to the Turing machines is needed [240, 239].

We also personally believe that all variants of the Church-Turing Thesis above are actually about "*reasonableness*": what is a reasonable notion of mechanical method in mathematics and logic for the Church-Turing thesis, what is a reasonable machine for Thesis M, or what is a reasonable model of the physics of our world for the last variant. In a contrapositive way, establishing a hyper computability result is a way to outline what should (possibly) be corrected in the model to make it reasonable [63].

Notice that this has been used for example to advocate for corrections of models from physics by Warren D. Smith in several papers [311, 312], taking the Church-

Turing thesis as a postulate, or if one prefers in the context of physics, as a physical law of our physical world.

For accounts on physical limitations against hypercomputational possibilities, one can refer to [124].

6.4.4 Can Analog Machines Compute Faster?

All previous variants also have effective versions dealing with complexity, sometimes called the *Effective Church-Turing Thesis* or *Strong Church-Turing Thesis (SCTT)*. We now come to the question of whether analog machines satisfy these versions.

In 1986, one of the first papers devoted to the explicit question of whether analog machines can be more efficient than digital ones was published [327]. The authors claim that the SCTT is provably true for continuous-time dynamical systems described by any Lipschitzian ODE $y' = f(y)$. However, their proof assumes that the time variable remains bounded. No clear argument is established for the general case, and this was considered to be an open problem until very recently. Interesting results and discussions on the subject can also be found in [55].

The question of whether analog machines satisfy the SCTT turns out to be deeply related to the question on how resources such as time are measured. In short, the difficulty is that the naive idea of using the time variable of the ODE as a measure of "time complexity" is problematic, since time can be arbitrarily contracted in a continuous system due to the "Zeno phenomenon". For example, consider a continuous system defined by an ODE $y' = f(y)$ where $f : \mathbb{R} \to \mathbb{R}$ and $y : \mathbb{R} \to \mathbb{R}$. Now consider the system

$$\begin{cases} z' = f(z)u \\ u' = u \end{cases}$$

where $u, z : \mathbb{R} \to \mathbb{R}$. It is not difficult to see that this system rescales the time variable and that its solution is given by $u(t) = e^t$ and $z(t) = y(e^t)$. Therefore, the second ODE simulates the first, with an exponential acceleration.

In a similar manner, it is also possible to present an ODE which has a solution with a component $u : [0, \frac{\pi}{2}) \to \mathbb{R}$ such that $u(t) = y(\tan(t))$, i.e., it is possible to contract the whole real line into a bounded set. Thus any language computable by the first system (or, in general, by a continuous system) can be computed by another continuous system in time $O(1)$. This problem has been observed or used in many continuous models [294, 295, 256, 61, 62, 12, 100, 142, 130, 131].

Notice that in addition to time contraction, space contraction is also possible, by considering changes on space variables. Note however, that space contractions are more model-dependent and are not always possible. For example, one could consider the system $z(t) = y(t)e^{-t}$, which makes the system exponentially smaller. However, doing so with a polynomial differential equation (for example) requires one to add a variable $u(t) = e^t$ to the system, thus making the contraction less useful, if useful at all.

Such time or space contraction phenomena seem, in many systems, to physically correspond to some infinite energy. However, *energy* is not a mathematically universally defined concept, and defining a robust notion of "time complexity" in those systems was an open problem until lately.

It has only recently been shown that analog systems satisfy the SCTT, if "time" (as in "time complexity") is measured by the length of the trajectory [72, 73], and if considering polynomial ordinary differential equations. Since this class of ODEs is very wide and covers in practice most reasonable classes, this can be considered to be a definitive proof of the statement that many analog systems satisfy the SCTT.

Note that this does not cover all variants of the SCTT, since ODEs do not cover all known physics. In particular, models that rely on quantum effects, General Relativity, and more generally models that rely on physical experiments (see Section 6.6.2 for example) do not fall into this class. In other words, this does not cover the "physical" variants of the SCTT.

6.4.5 Some Philosophical Aspects

Some of the previously mentioned results led to discussions in the context of philosophy.

First, the consideration of systems that could realize hypercomputation gives rise to the following philosophical question: since the physical construction of a hypercomputational model is outside the framework of computability theory, answering the *hypercomputation thesis* requires the construction of a physical model of hypercomputation. It has been argued that, even if such a model is built, it would be impossible to verify that this model is indeed computing a function non-computable by a Turing machine [169, 143]. However, it was pointed out that this objection is not specific to hypercomputation since it is already not possible to verify that a Turing computable function is correct in finite time in general [132]. Furthermore, simply because there is no systematic way of computing the values of the halting function does not mean that one could not (in principle) assign one mathematician to each value and ask each of them to come up with a different method, therefore bypassing the problem of uniformity of computation.

The possibility of new computational models which may be exponentially more efficient than any previously known machine (in a wide context, including for example models based on quantum mechanics) suggests alternative, empirical views of complexity theory, usually considered as part of logic and computer science. The arguments in favor of this view and its consequences have been discussed in [273, 274]. It raises some fundamental questions such as whether the empirical limits of computing are identical to limits of algorithms or, to put it differently, what is the capability of symbolic processes to simulate empirical processes. This makes it possible to study the epistemology of computations realized by a machine. In particular, the existence of a function computable by a machine but not by an algorithm

would imply that some mathematical problems are solvable by empirical processes without any mathematical method [272, 274].

The statement that there is no equivalent of the Church-Turing thesis for computation over the reals (often relying on the non-convergence of formalisms, as for discrete computations) is shown in [275] to confound the issue of the extension of the notion of effective computation to the reals on the one hand, and the simulation of analog computers by Turing machines on the other hand. It is possible in both cases to argue in favor of a Church-Turing thesis over the reals [275]. It has also been argued that the analog computation literature often mixes up the concepts of continuous-valued computations and analog machines, and that a concept called *model-based computation* can help to untangle the misconception, by offering a two-dimensional view of computation: one dimension concerns models and the other the type of variables that are used [42].

Reflections on whether nature can be considered as computing and even about the notion of computation have also been discussed [231] in light of new technologies in physics and new ideas in computer science such as quantum computing, networks, and non-deterministic algorithms.

Some original views on the Church-Turing thesis have also been expressed by Dowek [151, 150]. The statement that all physically realized relations can be expressed by a proposition in the language of mathematics (which can be called the *Galileo thesis*) can be considered as a consequence of the Physical Church Thesis [151].

6.5 Theory of Analog Systems

We now review the various computational theories that have been developed for analog systems.

6.5.1 Generic Formalizations of Analog Computations

There have been several approaches to formalize analog computations in a general setting.

6.5.1.1 Formalization by Abstract State Machine

A formalization of generic algorithms covering both analog and classical, discrete- and continuous-time algorithms has been proposed in [69]. This framework, extending [68], is based on an extension and reformulation of *abstract state machines* (ASM) from Gurevich [194]. The notion of an analog ASM program is introduced, and a completeness result is proven: any process satisfying the three postulates gen-

eralizing those of [194] is demonstrated to correspond to some analog ASM program [69]. This is intended to be a first step towards a formalization and a proof of an equivalent of a Church-Turing thesis for analog continuous- and discrete-time systems in the spirit of what has been achieved for discrete-time models [58, 146, 59].

6.5.1.2 Formalization by Fixed-Point Techniques

An approach to define computations by analog models in general is to consider networks of analog modules connected by channels [324], processing data from a metric space A, and operating with respect to a global continuous clock $\mathbb{T}$. The inputs and outputs of the network are continuous streams ($C = \mathbb{T} \to A$) and the input-output behavior of the network, usually with external parameters, is modeled by a function $\phi : C^r \times A^p \to C^q$. The authors focus on the important case where the modules are *causal*, that is the output at time t only depends on the inputs over $[0,t]$.

Equational specifications of such circuits, as well as their semantics, are given by fixed-points of operators over the space of continuous streams. Under suitable hypotheses, this operator is contracting and an extension of the Banach fixed-point theorem for metric spaces guarantees existence and uniqueness of the fixed point. Moreover, that fixed-point can also be proven to be continuous and *concretely* computable whenever the basic modules are.

A general framework to deal with fixed-point techniques in analog systems, based on Fréchet spaces, has recently been developed in [282].

An abstract model of computability over data streams using a high-level programming language, an extension of "while" programs, over abstract data-types (multi-sorted algebras) has been considered in [325]. The authors analyze when concrete and abstract models are equivalent.

6.5.1.3 Formalization and Proof Theory for Cyberphysical Systems

A whole theory for the analysis, logic and proofs for cyberphysical systems has been developed [281]. Rich logical theories to cover both continuous differential equation dynamics and discrete changes involved in cyberphysical or hybrid systems have been proposed and proven to be sound and complete [276, 278, 279]. These logical foundations, built incrementally in a series of articles, now form an elegant proof-theoretical bridge aligning the theory of continuous systems with the theory of discrete systems [277, 280]. They are the basis of theorem provers *KeYmaera* and *KeYmaera X*, which have demonstrated their impact on various concrete applications. For an overview of all the existing works along these lines of research at both theoretical and practical levels, refer to [280, 281].

6.5.2 $\mathbb{R}$-recursion Theory

$\mathbb{R}$-recursive functions were introduced as a theory of recursive functions on the reals built in analogy with classical recursion theory to deal with conceptual analog computers operating in continuous-time [256]. Moore considered the smallest class of functions, which is obtained from the constants $0, 1$ and the projections and which is closed under composition, integration and minimization. He demonstrated that various non-recursively enumerable sets fall into the proposed hierarchy, defined according to the number of minimizations involved.

The initial theory [256] suffers from some mathematical difficulties when considering several nested minimization operators, but its foundational ideas gave birth to several lines of research, both at the computability and complexity level.

At the computability level, since minimization is the operation that gives rise to uncomputable functions, and difficulties, a natural question is to ask whether it can be replaced by some other natural and better defined operator of mathematical analysis. This can be done by replacing minimization by some limit operation [260]. In addition, it is shown that the obtained hierarchy does not collapse [230, 229], which implies that infinite limits and first-order integration are not interchangeable operations [135]. A presentation and overview of the obtained real recursive function theory is presented in [136].

The algebra of functions built without the minimization operator only contains analytic functions and is not closed under iteration [104]. Closure by bounded products, bounded sums and bounded recursion has also been investigated.

However, if an arbitrarily smooth extension θ to the reals of the Heaviside function is included in the set of basic functions, then extensions to the reals of all primitive recursive functions are obtained. More generally, several authors studied extensions of the algebra of functions by considering an integration operator restricted to *linear* differential equations, and considering that such an arbitrarily smooth extension θ was among the basic functions. The obtained class contains extensions to the reals of all the elementary functions [101, 105, 102, 103]. By adding a suitable basic exponential iterate, extensions to the reals of all the functions of the classes of Grzegorczyk's hierarchy can also be obtained.

Instead of asking which computable functions over $\mathbb{N}$ have extensions to $\mathbb{R}$ in a given function algebra, one can consider classes of functions over $\mathbb{R}$ computable according to recursive analysis, and characterize them precisely with function algebras. This was done for elementarily computable functions [74], considering a restricted limit schema, and for computable functions [75], considering a restricted minimization schema.

A generic framework for presenting previous results, based on the notion of approximation, has been developed in [106].

At the complexity level, characterizations of complexity classes such as P and NP using $\mathbb{R}$-recursive algebra have been obtained [262]. This is done by ensuring that at every step of the construction of a function in previous classes, each component cannot grow faster than a quasi-polynomial. This allows one to transfer classical

questions from complexity theory to the context of real and complex analysis [135, 262, 261].

6.5.3 Analog Automata Theories

Several lines of research have been devoted to adapting classical discrete automata theory to analog domains or continuous-time.

Rabinovich and Trakhtenbrot have developed a *continuous-time automata* theory in [287, 322, 286]. In this approach, automata are not considered as language recognizers, but as computing operators on signals, that is on functions from the non-negative real numbers to a finite alphabet, viewed as states of a channel. For a presentation as well as extensions, see [170].

Another approach has been considered, where ODEs equipped with a tape-like function memory are used to recognize sets of piecewise continuous functions, yielding a Chomsky-like hierarchy [297].

A class of recognizable functions extending stochastic and quantum functions and with a cut point theorem similar to the one for probabilistic automata theory and with nice closure properties has been determined in [81].

Topological automata were introduced as a generalization of previous definitions of probabilistic and quantum automata [212]: deterministic or nondeterministic probabilistic and quantum automata are proven to recognize only regular languages with an isolated threshold. A topological theory of continuous-time automata has also been developed [247]: the basic idea is to replace finiteness assumptions in the classical theory of finite automata by compactness assumptions. Some existence results and a Myhill-Nerode theorem are obtained in this framework covering both finite automata and continuous automata.

Generalized finite automata working over arbitrary structures have been introduced [173]. Structural properties of accepted sets as well as computational hardness of classical questions for this model have been investigated [244]. A restricted version of the model has also been considered [245]: it is demonstrated to satisfy a pumping lemma, and to yield decidability results closer to what would be expected for a notion of finite automata over arbitrary structures, such as the reals.

The abovementioned approaches are not easily connected, and are rather independent views and models.

6.6 Analyzing the Power and Limitations of Analog Computations

6.6.1 Neural Networks

The computational power of artificial neural networks has been investigated intensively in the 1990's in many papers: see the instructive survey [309] and more recently [91]. Finite analog recurrent networks with integer weights are known to be essentially equivalent to finite automata, while finite analog recurrent networks with rational weights can simulate arbitrary Turing machines step by step [307]. When considering arbitrary real weights (possibly non-computable) one obtains the non-uniform versions of the associated complexity and computability classes. In particular, one gets models that can decide predicates or sets that are not computable by Turing machines [306]. For a full discussion, see [305]. Improvements on the mapping from a Turing machine to a recurrent artificial neural network have recently been proposed [110]. It has also been proven that the question $P = ?NP$ relativizes naturally to similar questions related to artificial neural networks with real weights [134].

Previous results have been extended to the case of evolving recurrent neural networks, i.e., networks where the synaptic weights can be updated at each discrete time step [92, 94]. Interactive recurrent neural networks have been considered in [93, 96].

When subjected to some infinite binary input stream, the Boolean output cells necessarily exhibit some attractor dynamics. A characterization of the expressive power of several models of Boolean, sigmoidal deterministic, and sigmoidal nondeterministic first-order recurrent neural networks in relation to their attractor dynamics has been established in [98, 97, 99]. This extends results about their expressive power characterized in terms of topological classes in the Borel hierarchy over the Cantor space [95].

The computational power of spiking neuron models has been investigated in a series of papers: refer to [309, 233, 177] for surveys.

6.6.2 Physical Oracles

In a recent series of papers, started by [44], Beggs, Costa, Poças and Tucker consider various dynamical systems doing computations involving discrete data and physical experiments. Such systems can be modeled by Turing machines with oracles that correspond to physical processes. A good summary of their results can be found in [15]. They analyze the computational power of these machines and more specifically the limits of polynomial-time computations. They show that for a broad class of physical oracles, an upper bound on polynomial-time computations is the

non-uniform complexity class[2] BPP//log$\star$, and that P/poly can be attained by using non-computable analog-digital interface protocols.

Without giving too many details, in this model the Turing machine has access to a physical oracle, such as [45]. Whenever the machine wants to call the oracle, it writes some input on the oracle tape and enters a special oracle state. The machine is then suspended and a physical experiment starts. Importantly, a time schedule function T, usually a time-constructible function, is associated with each machine. If the experiment finishes in time less than $T(|z|)$, where z is the content of the oracle tape, the machine resumes in one of finitely many states encoding the outcome of the experiment (typically *YES* or *NO*). Otherwise, the experiment is stopped and the machine resumes in a special state *TIMEOUT*.

The authors analyzed many kinds of physical oracles and identified three major forms of experiments. In most cases, the experiment encodes a single real number a and the oracle tape encodes a rational x. Each class of experiments specifies not only the behavior of the experiment but also how long it takes to complete.

- **Two-Sided Case:** the experiment takes time $t(x,a) = C/|x-a|^d$ for $x \neq a$ and answers YES if $x < a$ and NO if $x > a$. It always times out if $a = x$. A typical example is a scale, where the time needed to detect movement to either side is inversely exponential to the difference between the masses.
- **Threshold Case:** the experiment takes time $t(x,a) = C/|x-a|^d$ for $x > a$ and always answers YES. It always times out if $x \leqslant a$. Several examples including the photoelectric effect and Rutherford scattering are given by the authors.
- **Vanishing Case:** the experiment can only detect that $x \neq a$ and times out if $x = a$. However, given two experiments x, x', we can detect which of x and x' is closer to a. Informally, the reader can think of an experiment where we can measure $\alpha(x-a)^2$ (with α some unknown parameter): we cannot detect the sign of $x-a$ but for two experiments x, x', α is the same so we can compare $\alpha(x-a)^2$ to $\alpha(x'-a)^2$. Examples include some modified Wheatstone bridge and Brewster's angle experiments.

Orthogonally to the type of physical experiment, the authors also study variants on how precise the experiment is with respect to x. In the *infinite*-precision case, the experiment is done exactly with x. In the *unbounded*-precision case, the experiment is done with $x' \in [x-\varepsilon, x+\varepsilon]$ (drawn uniformly at random and experiments are independent) where ε is part of the input and can be arbitrarily small (but not zero). In the *fixed*-precision case, ε is a constant of the experiment and cannot be changed.

[2] Let $\mathscr{B}$ be a class of sets and $\mathscr{F}$ a class of functions. The advice class $\mathscr{B}/\mathscr{F}$ is the class of sets A such that there exists $B \in \mathscr{B}$ and $f \in \mathscr{F}$ such that for every word w, $w \in A$ if and only if $\langle w, f(|w|)\rangle \in B$. We use *log* to denote the class of functions f such that $|f(n)| = \mathscr{O}(\log n)$ as $n \to \infty$. We use *poly* to denote the class of functions f such that $|f(n)| = \mathscr{O}(n^k)$ for some k as $n \to \infty$. Thus $P/poly$ corresponds to a non-uniform polynomial-size advise class. The class $BPP//\mathscr{F}\star$ is the class of sets A such that there is a probabilistic polynomial-time Turing machine $\mathscr{M}$, a prefix function $f \in \mathscr{F}\star$ (i.e., $f(n)$ is a prefix of $f(n+1)$) and a constant $\gamma < \frac{1}{2}$ such that for every word w, on input $\langle w, f(|w|)\rangle$, $\mathscr{M}$ rejects with probability at most γ if $w \in A$, and accepts with probability at most γ if $w \notin A$. See [44] for more details on those classes.

More recently, the authors have developed a hierarchy for BPP//log$\star$ based on counting calls to a non-deterministic physical oracle modeled by a random walk on a line [43].

A general framework for describing physical computations covering the above approaches has been proposed in [332].

6.6.3 On the Effect of Noise on Computations

The invariant (ergodic) measure of dynamical systems has been proven to remain computable even if a small amount of noise is introduced in the system [83]. This demonstrates that while some dynamical systems have been proven to simulate Turing machines, the presence of noise forbids incomputability. Actually, when considering a compact domain, it has been proven that analog neural nets with Gaussian or other common noise distributions cannot even recognize arbitrary regular languages [234, 235].

Some tight bounds on the space complexity of computing the ergodic measure of a low-dimensional discrete-time dynamical systems affected by Gaussian noise have been obtained [84].

Previous results have implied some limitations on the ability of physical systems to perform computations, namely on their ability to store information. Memory is roughly defined as the maximal amount of information that the evolving system can carry from one instance to the next, and is demonstrated not to be able to grow too fast.

This offers support for the following postulate for physical computations. If a physical system has memory s available to it, then it is only capable of performing computations in the complexity class $SPACE(poly(s))$ even when provided with unlimited time [85]. This has been proven to hold for several classes of dynamics and noise models over a compact domain.

6.6.4 Complexity Theories for Analog Computations

There have been several attempts to build a complexity theory for continuous-time systems (but not intended to cover generic ODEs).

This includes the theory where the global minimizers of particular energy functions are supposed to give solutions of some problem [182, 183]. The structure of such energy functions leads to the introduction of problem classes U and NU, with the existence of complete problems for these classes.

Another attempt is focused on a very specific type of systems: *dissipative flow models* [49]. This theory has been used in several papers from the same authors to study a particular class of flow dynamics [48] for solving linear programming problems.

Neither of the previous two approaches is intended to cover generic ODEs, and neither of them is able to relate the obtained classes to classical classes of computational complexity, unlike the approach presented in Section 6.7.

6.6.5 Chemical Reaction Networks

Recent years have seen important publications on the computational power of *chemical reaction networks* (CRNs) [314, 129]. The model is built by analogy with concrete chemistry, see also [289] for a presentation. A program corresponds to a set of chemical rules over a finite set of species, abstracting from the matter conservation laws, considering well-mixed solutions, and ignoring spatial properties of the molecules. They can be seen as natural extensions of the population protocols discussed in Section 6.2.8: here reactions are not assumed to be between pairs of agents/molecules, and reaction rates can possibly be considered.

One can distinguish various dichotomies of computations in CRNs [140, 313]: the number of molecules can be considered to be either *discrete* or *continuous*. Computations can be considered either *uniform* (the same CRN handles all inputs) or *non-uniform* (for each input size, a different CRN is considered). In the *deterministic* setting the output is considered to be guaranteed, while in the *probabilistic* setting some probability of error is considered. The *halting approach* considers notions of acceptance where agents irreversibly produce an answer, while in the *stabilizing approach* the population eventually stabilizes to an answer, in the spirit of the acceptance criteria for population protocols.

CRNs have clear connections to various other models, including Petri nets and vector addition systems [129]. These models can be classified as *unordered* with respect to classical models of computability such as Turing machines where an order on instructions is considered. The presence of reaction rates or probabilities is a way to provide back some order on instructions, while forgetting order in classical models of computability gives rise to similar models [129].

This abstract model is also motivated by concrete implementations using an experimental technique called DNA strand displacement [315, 107, 285, 120]. Implementations with proteins are also considered [164].

The computational power of CRNs over many of these variants has been studied in several papers [314, 118, 140]. It has been proven, using in particular the strong technical result from [160], that chemical reaction networks turn out to be a robust model from a computability point of view [87] in the sense that variations on the notion of output conventions for error-free computations basically yield either Turing machines or a model whose power is equivalent to the population protocols discussed in Section 6.2.8, that is to say whose computational power is limited to semi-linear sets.

Lower bounds on the time required to build rules that would act as a “timer” have been obtained [149], answering the open question on the ability of population

protocols to perform a fast leader election in the probabilistic settings, as well as their ability to simulate arbitrary algorithms from a uniform initial state.

The case of the continuous setting with the usual natural semantics has been an open problem before being solved very recently [164] using the theory discussed in Section 6.7. The computational power of a variant where reaction rates are abstracted was previously characterized [119].

6.7 Computations by Polynomial Ordinary Differential Equations

Polynomial ordinary differential equations have been shown to be closely related to the GPAC of Shannon discussed in Section 6.2.2.1 and to have a very rich theory. In many respects, they may now be considered to play an equivalent role in continuous-time models to that of Turing machines in discrete-time models.

6.7.1 GPAC and Polynomial Ordinary Differential Equations

The GPAC, presented in Section 6.2.2.1, can also be presented in terms of polynomial ordinary differential equations. More precisely, y is *generable by a GPAC* if and only if there exist functions $z = (z_1, \ldots, z_n)$, a vector of polynomials p and an initial condition z_0 such that

$$y \equiv z_1, \qquad z(0) = z_0, \quad \text{and} \quad z' = p(z).$$

In other words, y is a component of the solution of a system of polynomial differential equations. Advantages of this presentation are that it eliminates all the problems related to the domain of definition and that the solution is necessarily unique. In particular, the solutions are always analytic functions.

This fact actually has a complicated history because of gaps in the proof found by various authors and the refinements in the definition of the GPAC: Shannon, while introducing his model [303], proved that any function y generated by a GPAC is a solution of a polynomial Differential Algebraic Equation (pDAE), that is there exists a polynomial p such that

$$p(y, y', \ldots, y^{(n)}) = 0 \tag{6.3}$$

for some integer n. The converse inclusion was claimed by Shannon, which led some authors to define the GPAC directly in terms of differential equations [284]. However this equivalence requires some care because of the domain of definition, and furthermore, DAEs do not necessarily have a unique solution, which complicates this approach. It was later realized that unary functions generated by a GPAC

(with some restrictions to fix earlier problems) are exactly components of polynomial Initial Value Problems (pIVP) [186]. Several variations on GPAC circuits were explored and proven to be all equivalent, which essentially shows that the above notion is robust and probably the right definition to consider [184].

A famous statement, due to Rubel [292], shows that polynomial Differential Algebraic Equations (pDAEs) are *universal* for continuous functions. More precisely, there is a universal pDAE of order 4 whose solutions can approximate any continuous function with arbitrary precision [292]. This work has recently been improved [139] to hold for order 3. It has been established very recently that this also holds for polynomial ordinary differential equations [76].

6.7.2 GPAC Generable Functions

The class of generable functions is particularly robust because if f and g are generable then $f \pm g$, fg, $\frac{f}{g}$ and $f \circ g$ are also generable. Moreover, if f is generable and y satisfies the differential equation $y' = f(y)$ then y is also generable. Shannon's original work was intended for circuits with several inputs, but this was never formalized. Recent work provides a proper description of the class of generable functions over multi-dimensional domains and shows that it also enjoys many closure properties [71].

6.7.3 GPAC Computability

The fact that the functions generated by the GPAC (i.e., generable functions) are analytic (or C^∞ for the more relaxed models, without external inputs) has historically been seen as a limitation of the GPAC.

This has been proven to be an artifact of the model and an alternative notion of computability based on polynomials ODEs has been proposed [65]. In fact the authors prove that the GPAC and Turing machines are equivalent. The fundamental idea is that while generable functions correspond to "*real-time computability*" (i.e., the system computes the answer instantaneously), considering a notion similar to the one used for Turing machines (i.e., the system evolves and "converges effectively" to the answer) yields a computational power equivalent to that of Turing machines.

This equivalence between the GPAC and Computable Analysis can be reformulated as follows.

Theorem 6.7.1 (Equivalence between GPAC and Computable Analysis, [65]). *A function $f : [a,b] \to \mathbb{R}$ is computable (in the sense of Computable Analysis) if and only if there exists an integer d and a vector of polynomials $p : \mathbb{R}^d \to \mathbb{R}^d$ with rational coefficients such that for any $x \in [a,b]$, the unique solution $y : \mathbb{R}_{\geqslant 0} \to \mathbb{R}^d$ to*

$$y(0) = (0,\dots,0,x), \qquad y'(t) = p(y(t))$$

satisfies that

$$|f(x) - y_1(t)| \leqslant y_2(t) \quad \textit{and} \quad y_2(t) \to 0 \textit{ as } t \to \infty.$$

In other words, a function f is computable if there is a GPAC that, on input x, has one of its components (y_1) converge to $f(x)$. Not only does it converge, but another component (y_2) gives a bound on the error between $f(x)$ and y_1.

This unexpected connection between Turing computability and the GPAC was recently refined [72, 73] at the level of complexity, with a characterization of the class P and polynomial-time computable real functions.

A difficult point in the context of analog computability, and in the GPAC in particular, is to define a notion of complexity that makes sense and is sufficiently robust: see discussions in Section 6.4.4. In fact, the intuitive notion of measuring the complexity based on the convergence rate (i.e., how fast $y_1(t) \to f(x)$) does not work.

One recently proposed solution to this problem is to measure the complexity using the *length of the curve y* instead of time. This process is illustrated in Figure 6.4. We recall that the length of a curve $y \in C^1(I, \mathbb{R}^n)$ defined over some interval $I = [a,b]$ is given by $\text{len}_y(a,b) = \int_I \|y'(t)\|_2 \, \mathrm{d}t$,

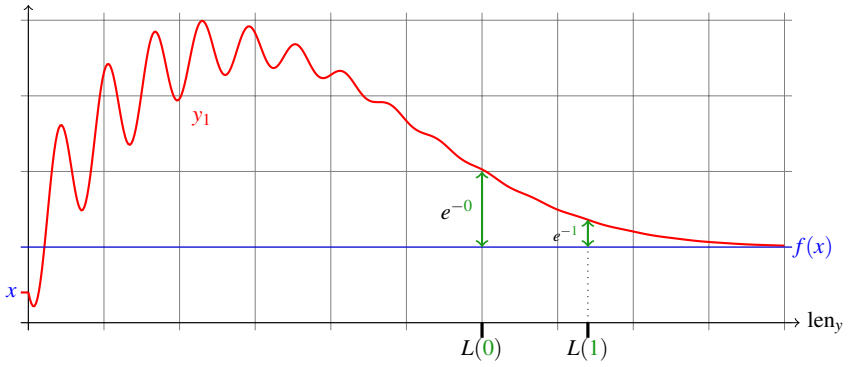


Fig. 6.4 Graphical representation of analog computability (Theorem 6.7.2): on input x, starting from initial condition $(x,0,\ldots,0)$, the polynomial ordinary differential equation $y' = p(y)$ ensures that $y_1(t)$ gives $f(x)$ with accuracy better than $e^{-\mu}$ as soon as the length of y (from 0 to t) is greater than $L(\mu)$. Note that we did not plot the other variables $y_2, \ldots, y_d$ and the horizontal axis measures the length of y (instead of the time t).

Theorem 6.7.2 (Equivalence between GPAC and CA (Complexity), [73]). *A function $f : [a,b] \to \mathbb{R}$ is computable in polynomial-time (in the sense of Computable Analysis) if and only if there exists a polynomial $L : \mathbb{R}_{\geqslant 0} \to \mathbb{R}_{\geqslant 0}$, an integer d and a vector of polynomials $p : \mathbb{R}^d \to \mathbb{R}^d$ with coefficients in $\mathbb{Q}$, such that for any $x \in [a,b]$, the unique solution $y : \mathbb{R}_{\geqslant 0} \to \mathbb{R}^d$ to*

$$y(0) = (x,0,\ldots,0), \qquad y'(t) = p(y(t))$$

satisfies for all $t \in \mathbb{R}_{\geqslant 0}$*:*

- *for any* $\mu \in \mathbb{R}_{\geqslant 0}$*, if* $\mathrm{len}_y(0,t) \geqslant L(\mu)$ *then* $|f(x) - y_1(t)| \leqslant e^{-\mu}$,
- $\|y'(t)\| \geqslant 1$.

In other words, the precision of y_1 increases with the length of the curve. More precisely, as soon as the length between 0 and t is at least $L(\mu)$, the precision is at least $e^{-\mu}$. Notice how rescaling the curve would not help here since it does not change the length of y. The second condition on the derivative of y prevents some pathological cases and ensures that the curve has infinite length, and thus that y_1 indeed converges to $f(x)$. It is possible to extend this equivalence to multivariate functions and unbounded input domains such as $\mathbb{R}$, by making L take into account the norm of x.

It also possible to define the class P directly in terms of differential equations, by encoding words with rational numbers. Again the length plays a crucial role, but since a differential equation does not "stop", the component y_1 is used to signal that it accepts ($y_1 \geqslant 1$) or rejects ($y_1 \leqslant -1$). Figure 6.5 illustrates this process.

Theorem 6.7.3 (Analog characterization of P, [73]). *A language* $\mathscr{L} \subseteq \{0,1\}^*$ *belongs to P, the class of polynomial-time decidable languages, if and only if there exist a polynomial* $L : \mathbb{N} \to \mathbb{N}$*, an integer* d *and a vector of polynomials* $p : \mathbb{R}^d \to \mathbb{R}^d$ *with coefficients in* $\mathbb{Q}$*, such that for all words* $w \in \{0,1\}^*$*, the unique solution* $y : \mathbb{R}_{\geqslant 0} \to \mathbb{R}^d$ *to*

$$y(0) = (0, \ldots, 0, |w|, \psi(w)), \qquad y'(t) = p(y(t))$$

where $\psi(w) = \sum_{i=1}^{|w|} w_i 2^{-i}$*, satisfies for all* $t \in \mathbb{R}_{\geqslant 0}$*:*

- *if* $|y_1(t)| \geqslant 1$ *then* $|y_1(u)| \geqslant 1$ *for all* $u \geqslant t \geqslant 0$ *(and similarly for* $|y_1(t)| \leqslant -1$*),*
- *if* $w \in \mathscr{L}$ *(resp.* $\notin \mathscr{L}$*) and* $\mathrm{len}_y(0,t) \geqslant L(|w|)$ *then* $y_1(t) \geqslant 1$ *(resp.* $\leqslant -1$*),*
- $\|y'(t)\| \geqslant 1$.

One clear interest of the previous statements is that they provide a way to define classical concepts from the theory of computation (computable function, polynomial-time computable functions) only using concepts from analysis, namely polynomial ordinary differential equations.

Acknowledgements We would like to thank several people for various feedback, and many helpful comments: Andrew Adamatzky, Amir Ali Amhadi, Cameron Beebe, Jérémie Cabessa, Liesbeth Demol, Dave Doty, Laurent Doyen, Jérôme Durand-Lose, Marie-Jo Durand-Richard, Javier Esparza, François Fages, Goran Frehse, Laurent Fribourg, Daniel Graça, Eric Goubault, Emmanuel Hainry, Emmanuel Jeandel, Manuel Lameiras Campagnolo, Jérôme Leroux, Jack Lutz, Wolfgang Maass, Klaus Meer, Pekka Orponen, André Platzer, Maël Pégny, Cristóbal Rojas, Keijo Ruohonen, Zoltan Toroczkai, John Tucker, Bernd Ulmann, Erik Winfree, Jeffery Zucker, as well as two anonymous referees.

Special thanks to Ulf Hashagen for many references related to the history of computing, Andrew Adamatzky and Keijo Ruohonen for several references and comments, sometimes repeated literally in this document, about some physical or unconventional computing models.

Many thanks to Jérémie Cabessa, Jérôme Durand-Lose and Klaus Meer for helpful feedback including typographic issues regarding a preliminary version of this document.

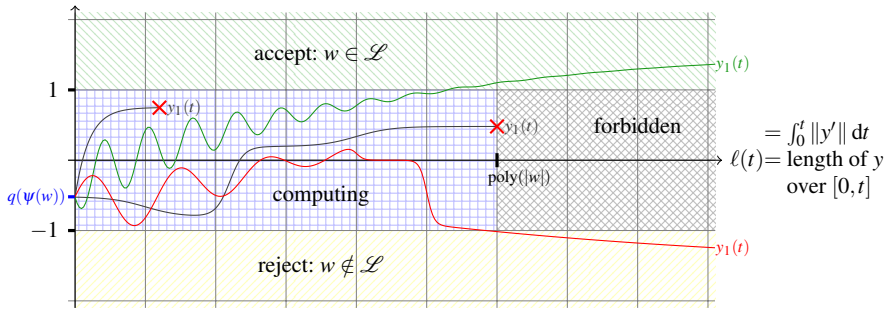


Fig. 6.5 Graphical representation of the analog characterization of P (Theorem 6.7.3). The green trajectory represents an accepting computation, the red a rejecting one, and the gray are invalid computations. An invalid computation is a trajectory that is too slow (thus violating the technical condition), or that does not accept/reject in polynomial length. Note that we only represent the first component of the solution; the other components can have arbitrary behaviors.

References

1. http://www.analogmuseum.org/
2. https://www.tensorflow.org/
3. Aaronson, S., Watrous, J.: Closed timelike curves make quantum and classical computing equivalent. Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences **465**(2102), 631–647 (2009)
4. Adamatzky, A.: Physarum machine: implementation of a Kolmogorov-Uspensky machine on a biological substrate. Parallel Processing Letters **17**(04), 455–467 (2007)
5. Adamatzky, A.: Physarum Machines: Computers From Slime Mould, *World Scientific Series on Nonlinear Science Series A*, vol. 74. World Scientific (2010)
6. Adamatzky, A.: Atlas of Physarum Computing. World Scientific (2015)
7. Adamatzky, A., Costello, B.D.L., Asai, T.: Reaction-Diffusion Computers. Elsevier (2005)
8. Adleman, L.M.: Molecular computation of solutions to combinatorial problems. Science **266**, 1021–1024 (1994)
9. Ahmadi, A.A., Jungers, R.M.: Switched stability of nonlinear systems via SOS-convex Lyapunov functions and semidefinite programming. In: Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on, pp. 727–732. IEEE (2013)
10. Ahmadi, A.A., Jungers, R.M.: On complexity of Lyapunov functions for switched linear systems. IFAC Proceedings Volumes **47**(3), 5992–5997 (2014)
11. Ahmadi, A.A., Majumdar, A., Tedrake, R.: Complexity of ten decision problems in continuous time dynamical systems. In: American Control Conference (ACC), 2013, pp. 6376–6381. IEEE (2013)
12. Alur, R., Dill, D.L.: Automata for modeling real-time systems. In: M. Paterson (ed.) Automata, Languages and Programming, 17th International Colloquium, ICALP90, Warwick University, England, July 16-20, 1990, Proceedings, *Lecture Notes in Computer Science*, vol. 443, pp. 322–335. Springer (1990)
13. Alur, R., Dill, D.L.: A theory of timed automata. Theoretical Computer Science **126**(2), 183–235 (1994)
14. Alur, R., Henzinger, T.A., Lafferriere, G., Pappas, G.J.: Discrete abstractions of hybrid systems. Proceedings of the IEEE **88**(7), 971–984 (2000)
15. Ambaram, T., Beggs, E., Costa, J.F., Poças, D., Tucker, J.V.: An analogue-digital model of computation: Turing machines with physical oracles. In: A. Adamatzky (ed.) Advances in Unconventional Computing, pp. 73–115. Springer (2017)

16. Ames, A.D., Sastry, S.: Characterization of Zeno behavior in hybrid systems using homological methods. In: Proceedings of the 2005 American Control Conference, pp. 1160–1165. IEEE (2005)
17. Ames, A.D., Zheng, H., Gregg, R.D., Sastry, S.: Is there life after Zeno? Taking executions past the breaking (Zeno) point. In: 2006 American Control Conference, pp. 6–pp. IEEE (2006)
18. Andréka, H., Madarász, J.X., Németi, I.: Logical axiomatizations of space-time. samples from the literature. In: A. Prékopa, E. Molnár (eds.) Non-Euclidean Geometries: János Bolyai Memorial Volume, pp. 155–185. Springer US, Boston, MA (2006)
19. Angluin, D., Aspnes, J., Chan, M., Fischer, M.J., Jiang, H., Peralta, R.: Stably computable properties of network graphs. In: V.K. Prasanna, S. Iyengar, P. Spirakis, M. Welsh (eds.) Distributed Computing in Sensor Systems: First IEEE International Conference, DCOSS 2005, Marina del Rey, CA, USA, June/July, 2005, Proceedings, *Lecture Notes in Computer Science*, vol. 3560, pp. 63–74. Springer-Verlag (2005)
20. Angluin, D., Aspnes, J., Diamadi, Z., Fischer, M.J., Peralta, R.: Computation in networks of passively mobile finite-state sensors. In: Twenty-Third ACM Symposium on Principles of Distributed Computing, pp. 290–299. ACM Press (2004)
21. Angluin, D., Aspnes, J., Eisenstat, D., Ruppert, E.: The computational power of population protocols. Distributed Computing **20**(4), 279–304 (2007)
22. Angluin, D., Aspnes, J., Fischer, M.J., Jiang, H.: Self-stabilizing population protocols. ACM Transactions on Autonomous and Adaptive Systems (TAAS) **3**(4), 13 (2008)
23. Anstreicher, K.M.: Linear programming and the Newton barrier flow. Mathematical Programming **41**(1-3), 367–373 (1988)
24. Asarin, E.: Chaos and undecidabilty (draft) (1995), avalaible at http://www.liafa.jussieu.fr/~asarin/
25. Asarin, E.: Equations on timed languages. In: T.A. Henzinger, S. Sastry (eds.) Hybrid Systems: Computation and Control, First International Workshop, HSCC'98, Berkeley, California, USA, April 13-15, 1998, Proceedings, *Lecture Notes in Computer Science*, vol. 1386, pp. 1–12. Springer (1998)
26. Asarin, E., Caspi, P., Maler, O.: A Kleene theorem for timed automata. In: Proceedings, 12th Annual IEEE Symposium on Logic in Computer Science, pp. 160–171. IEEE Computer Society Press, Warsaw, Poland (1997)
27. Asarin, E., Caspi, P., Maler, O.: Timed regular expressions. Journal of the ACM **49**(2), 172–206 (2002)
28. Asarin, E., Dima, C.: Balanced timed regular expressions. Electronic Notes in Theoretical Computer Science **68**(5), 16–33 (2002)
29. Asarin, E., Maler, O.: Achilles and the tortoise climbing up the arithmetical hierarchy. Journal of Computer and System Sciences **57**(3), 389–398 (1998)
30. Asarin, E., Maler, O., Pnueli, A.: Reachability analysis of dynamical systems having piecewise-constant derivatives. Theoretical Computer Science **138**(1), 35–65 (1995)
31. Asarin, E., Mysore, V.P., Pnueli, A., Schneider, G.: Low dimensional hybrid systems–decidable, undecidable, don't know. Information and Computation **211**, 138–159 (2012)
32. Aspnes, J., Ruppert, E.: An introduction to population protocols. In: Bulletin of the EATCS, vol. 93, pp. 106–125 (2007)
33. Baartse, M., Meer, K.: Real interactive proofs for vpspace. In: LIPIcs-Leibniz International Proceedings in Informatics, vol. 58. Schloss Dagstuhl-Leibniz-Zentrum für Informatik (2016)
34. Baartse, M., Meer, K.: An algebraic proof of the real number PCP theorem. Journal of Complexity **40**, 34–77 (2017)
35. Basu, S.: A complex analogue of Toda's theorem. Foundations of Computational Mathematics **12**(3), 327–362 (2012)
36. Basu, S., Zell, T.: Polynomial hierarchy, Betti numbers, and a real analogue of Toda's theorem. Foundations of Computational Mathematics **10**(4), 429–454 (2010)
37. Bayer, D., Lagarias, J.C.: Karmarkar's linear programming algorithm and Newton's method. Mathematical Programming **50**(1-3), 291–330 (1991)

38. Bazille, H., Bournez, O., Gomaa, W., Pouly, A.: On the complexity of bounded time and precision reachability for piecewise affine systems. Theoretical Computer Science **735**, 132–146 (2018)
39. Beal, J., Dulman, S., Usbeck, K., Viroli, M., Correll, N.: Organizing the aggregate: Languages for spatial computing. corr, abs/1202.5509, 2012
40. Beauquier, D.: Pumping lemmas for timed automata. In: M. Nivat (ed.) Foundations of Software Science and Computation Structure, First International Conference, FoSSaCS'98, Held as Part of the European Joint Conferences on the Theory and Practice of Software, ETAPS'98, Lisbon, Portugal, March 28 - April 4, 1998, Proceedings, *Lecture Notes in Computer Science*, vol. 1378, pp. 81–94. Springer (1998)
41. Becchetti, L., Bonifaci, V., Dirnberger, M., Karrenbauer, A., Mehlhorn, K.: Physarum can compute shortest paths: convergence proofs and complexity bounds. In: F.V. Fomains, R. Freivalds, M. Kwiatkowska, D. Peleg (eds.) International Colloquium on Automata, Languages, and Programming, *Lecture Notes in Computer Science*, vol. 7966, pp. 472–483. Springer (2013)
42. Beebe, C.: Model-based computation. Natural Computing **17**, 271–281 (2017)
43. Beggs, E., Cortez, P., Costa, J.F., Tucker, J.V.: A hierarchy for BPP//log$\star$ hased on counting calls to an oracle. In: A. Adamatzky (ed.) Emergent Computation: A Festschrift for Selim G. Akl, pp. 39–56. Springer International Publishing, Cham (2017)
44. Beggs, E., Costa, J., Loff, B., Tucker, J.: Computational complexity with experiments as oracles. Proceedings of the Royal Society of London-A **464**(2098), 2777 (2008)
45. Beggs, E.J., Tucker, J.V.: Experimental computation of real numbers by Newtonian machines. In: Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, vol. 463, pp. 1541–1561. The Royal Society (2007)
46. Bell, P.C., Chen, S., Jackson, L.: On the decidability and complexity of problems for restricted hierarchical hybrid systems. Theoretical Computer Science **652**, 47–63 (2016)
47. Ben-Amram, A.M.: Mortality of iterated piecewise affine functions over the integers: Decidability and complexity. In: STACS, pp. 514–525 (2013)
48. Ben-Hur, A., Feinberg, J., Fishman, S., Siegelmann, H.T.: Probabilistic analysis of a differential equation for linear programming. Journal of Complexity **19**(4), 474–510 (2003)
49. Ben-Hur, A., Siegelmann, H.T., Fishman, S.: A theory of complexity for continuous time systems. Journal of Complexity **18**(1), 51–86 (2002)
50. Bennett, C.H.: Logical reversibility of computation. IBM Journal Research Development **6**, 525–532 (1973)
51. Bennett, C.H.: The thermodynamics of computation—a review. International Journal of Theoretical Physics **21**(12), 905–940 (1982)
52. Benveniste, A., Bourke, T., Caillaud, B., Pouzet, M.: Non-standard semantics of hybrid systems modelers. Journal of Computer and System Sciences **78**(3), 877–910 (2012)
53. Berry, G., Boudol, G.: The chemical abstract machine. Theoretical Computer Science **96** (1992)
54. Bishop, C.M.: Neural networks for pattern recognition. Oxford University Press (1995)
55. Blakey, E.W.: A model-independent theory of computational complexity: from patience to precision and beyond. Ph.D. thesis. University of Oxford (2010)
56. Blum, L., Cucker, F., Shub, M., Smale, S.: Complexity and Real Computation. Springer, New York (1998)
57. Blum, L., Shub, M., Smale, S.: On a theory of computation and complexity over the real numbers: *NP*-completeness, recursive functions and universal machines. Bulletin of the American Mathematical Society **21**(1), 1–46 (1989)
58. Boker, U., Dershowitz, N.: The Church-Turing thesis over arbitrary domains. In: A. Avron, N. Dershowitz, A. Rabinovich (eds.) Pillars of Computer Science: Essays Dedicated to Boris (Boaz) Trakhtenbrot on the Occasion of His 85th Birthday, vol. 4800, pp. 199–229. Springer (2008)
59. Boker, U., Dershowitz, N.: Three paths to effectiveness. In: A. Blass, N. Dershowitz, W. Reisig (eds.) Fields of Logic and Computation, *Lecture Notes in Computer Science*, vol. 6300, pp. 135–146. Springer (2010)

60. Bonnet, J., Yin, P., Ortiz, M.E., Subsoontorn, P., Endy, D.: Amplifying genetic logic gates. Science **340**(6132), 599–603 (2013)
61. Bournez, O.: Achilles and the Tortoise climbing up the hyper-arithmetical hierarchy. Theoretical Computer Science **210**(1), 21–71 (1999)
62. Bournez, O.: Complexité algorithmique des systèmes dynamiques continus et hybrides. Ph.D. thesis. École Normale Supérieure de Lyon (1999)
63. Bournez, O.: How much can analog and hybrid systems be proved (super-)Turing. Applied Mathematics and Computation **178**(1), 58–71 (2006)
64. Bournez, O., Campagnolo, M.L.: A survey on continuous time computations. In: S.B. Cooper, B. Löwe, A. Sorbi (eds.) New Computational Paradigms. Changing Conceptions of What Is Computable, pp. 383–423. Springer-Verlag, New York (2008)
65. Bournez, O., Campagnolo, M.L., Graça, D.S., Hainry, E.: Polynomial differential equations compute all real computable functions on computable compact intervals. Journal of Complexity **23**(3), 317–335 (2007)
66. Bournez, O., Cohen, J., Rabie, M.: Homonym population protocols. In: A. Bouajjani, H. Fauconnier (eds.) Networked Systems. Third International Conference, NETYS 2015, Agadir, Morocco, May 13-15, 2015, Revised Selected Papers, *Lecture Notes in Computer Science*, vol. 9466, pp. 125–139. Springer (2015)
67. Bournez, O., Cohen, J., Rabie, M.: Homonym population protocols. Theory of Computing Systems **62**(5), 1318–1346 (2018)
68. Bournez, O., Dershowitz, N., Falkovich, E.: Towards an axiomatization of simple analog algorithms. In: M. Agrawal, S.B. Cooper, A. Li (eds.) Theory and Applications of Models of Computation - 9th Annual Conference, TAMC 2012, Beijing, China, May 16-21, 2012. Proceedings, *Lecture Notes in Computer Science*, vol. 7287, pp. 525–536. Spinger-Verlag (2012)
69. Bournez, O., Dershowitz, N., Néron, P.: Axiomatizing analog algorithms. In: A. Beckmann, L. Bienvenu, N. Jonoska (eds.) Pursuit of the Universal, *Lecture Notes in Computer Science*, vol. 9709, pp. 215–224. Springer, Switzerland (2016), 12th Conference on Computability in Europe, CiE 2016, Paris, France, June 27 - July 1, 2016
70. Bournez, O., Fraigniaud, P., Koegler, X.: Computing with large populations using interactions. In: B. Rovan, V. Sassone, P. Widmayer (eds.) Mathematical Foundations of Computer Science, MFCS'12, *Lecture Notes in Computer Science*, vol. 7464. Spinger-Verlag (2012)
71. Bournez, O., Graça, D., Pouly, A.: On the functions generated by the general purpose analog computer. Information and Computation **257**, 34–57 (2017)
72. Bournez, O., Graça, D.S., Pouly, A.: Polynomial time corresponds to solutions of polynomial ordinary differential equations of polynomial length: the general purpose analog computer and computable analysis are two efficiently equivalent models of computations. In: I. Chatzigiannakis, M. Mitzenmacher, Y. Rabani, D. Sangiorgi (eds.) 43rd International Colloquium on Automata, Languages, and Programming, *LIPIcs. Leibniz Int. Proc. Inform.*, vol. 55, pp. Art. No. 109, 15. Schloss Dagstuhl. Leibniz-Zent. Inform., Dagstuhl, Germany (2016)
73. Bournez, O., Graça, D.S., Pouly, A.: Polynomial time corresponds to solutions of polynomial ordinary differential equations of polynomial length. Journal of the ACM **64**(6), Art. 38, 76 (2017)
74. Bournez, O., Hainry, E.: Elementarily computable functions over the real numbers and *r*-sub-recursive functions. Theoretical Computer Science **348**, 130–147 (2005)
75. Bournez, O., Hainry, E.: Recursive analysis characterized as a class of real recursive functions. Fundamenta Informaticae **74**(4), 409–433 (2006)
76. Bournez, O., Pouly, A.: A universal ordinary differential equation. In: I. Chatzigiannakis, P. Indyk, F. Kuhn, A. Muscholl (eds.) 44th International Colloquium on Automata, Languages, and Programming, *LIPIcs. Leibniz Int. Proc. Inform.*, vol. 80, pp. Art. No. 116, 14. Schloss Dagstuhl. Leibniz-Zent. Inform., Dagstuhl, Germany (2017)
77. Bouyer, P., Fahrenberg, U., Larsen, K.G., Markey, N., Ouaknine, J., Worrell, J.: Model checking real-time systems. In: E. Clarke, T. Henzinger, H. Veith (eds.) Handbook of Model Checking, pp. 1001–1046. Springer, Cham (2018)

78. Bouyer, P., Petit, A.: Decomposition and composition of timed automata. In: J. Wiedermann, P. van Emde Boas, M. Nielsen (eds.) Automata, Languages and Programming, 26th International Colloquium, ICALP'99, Prague, Czech Republic, July 11-15, 1999, Proceedings, *Lecture Notes in Computer Science*, vol. 1644, pp. 210–219. Springer (1999)
79. Bouyer, P., Petit, A.: A Kleene/Büchi-like theorem for clock languages. Journal of Automata, Languages and Combinatorics **7**(2), 167–186 (2002)
80. Bowles, M.D.: U.S. technological enthusiasm and British technological skepticism in the age of the analog brain. IEEE Annals of the History of Computing **18**(4), 5–15 (1996)
81. Bozapalidis, S.: Extending stochastic and quantum functions. Theory of Computing Systems **36**(2), 183–197 (2003)
82. Brattka, V.: Computable selection in analysis. In: K.I. Ko, K. Weihrauch (eds.) Computability and Complexity in Analysis, *Informatik Berichte*, vol. 190, pp. 125–138. FernUniversität Hagen (1995), CCA Workshop, Hagen, August 19–20, 1995
83. Braverman, M., Grigo, A., Rojas, C.: Noise vs computational intractability in dynamics. In: Proceedings of the 3rd Innovations in Theoretical Computer Science Conference, pp. 128–141. ACM (2012)
84. Braverman, M., Rojas, C., Schneider, J.: Tight space-noise tradeoffs in computing the ergodic measure. Sbornik: Mathematics **208**(12), 1758–1783 (2017)
85. Braverman, M., Schneider, J., Rojas, C.: Space-bounded Church-Turing thesis and computational tractability of closed systems. Physical Review Letters **115**(9), 098,701 (2015)
86. Brihaye, T., Doyen, L., Geeraerts, G., Ouaknine, J., Raskin, J.F., Worrell, J.: On reachability for hybrid automata over bounded time. In: L. Aceto, M. Henzinger, J. Sgall (eds.) International Colloquium on Automata, Languages, and Programming, *Lecture Notes in Computer Science*, vol. 6756, pp. 416–427. Springer (2011)
87. Brijder, R., Doty, D., Soloveichik, D.: Robustness of expressivity in chemical reaction networks. In: Y. Rondelez, D. Woods (eds.) DNA Computing and Molecular Programming, *Lecture Notes in Computer Science*, vol. 9818, pp. 52–66. Springer (2016), international Conference on DNA-Based Computers
88. Brockett, R.W.: Smooth dynamical systems which realize arithmetical and logical operations. In: H. Nijmeijer, J.M. Schumacher (eds.) Three Decades of Mathematical Systems Theory, *Lecture Notes in Computer Science*, vol. 135, pp. 19–30. Springer (1989)
89. Brockett, R.W.: Dynamical systems that sort lists, diagonalize matrices, and solve linear programming problems. Linear Algebra and its Applications **146**, 79–91 (1991)
90. Bush, V.: The differential analyser. Journal of the Franklin Institute **212**(4), 447–488 (1931)
91. Cabessa, J.: Part I Computational capabilities of recurrent neural networks. part II Limit knowledge: A topological approach to epistemic game theory. Habilitation thesis. Université Panthéon-Assas – Paris II Laboratoire d'Économie Mathématique et de Microéconomie Appliquée (LEMMA) (2016)
92. Cabessa, J., Siegelmann, H.T.: Evolving recurrent neural networks are super-Turing. In: Neural Networks (IJCNN), The 2011 International Joint Conference on, pp. 3200–3206. IEEE (2011)
93. Cabessa, J., Siegelmann, H.T.: The computational power of interactive recurrent neural networks. Neural Computation **24**(4), 996–1019 (2012)
94. Cabessa, J., Siegelmann, H.T.: The super-Turing computational power of plastic recurrent neural networks. International Journal of Neural Systems **24**(08), 1450,029 (2014)
95. Cabessa, J., Villa, A.E.: The expressive power of analog recurrent neural networks on infinite input streams. Theoretical Computer Science **436**, 23–34 (2012)
96. Cabessa, J., Villa, A.E.: The super-Turing computational power of interactive evolving recurrent neural networks. In: V. Mladenov, P. Koprinkova-Hristova, G. Palm, A.E.P. Villa (eds.) Artificial Neural Networks and Machine Learning, *Lecture Notes in Computer Science*, vol. 8131, pp. 58–65. Springer (2013), international Conference on Artificial Neural Networks - ICANN 2013
97. Cabessa, J., Villa, A.E.: An attractor-based complexity measurement for Boolean recurrent neural networks. PLoS One **9**(4), e94,204 (2014)

98. Cabessa, J., Villa, A.E.: Computational capabilities of recurrent neural networks based on their attractor dynamics. In: Neural Networks (IJCNN), 2015 International Joint Conference on, pp. 1–8. IEEE (2015)
99. Cabessa, J., Villa, A.E.: Expressive power of first-order recurrent neural networks determined by their attractor dynamics. Journal of Computer and System Sciences **82**(8), 1232–1250 (2016)
100. Calude, C.S., Pavlov, B.: Coins, quantum measurements, and Turing's barrier. Quantum Information Processing **1**(1-2), 107–127 (2002)
101. Campagnolo, M.L.: Computational complexity of real valued recursive functions and analog circuits. Ph.D. thesis. IST, Universidade Técnica de Lisboa (2001)
102. Campagnolo, M.L.: The complexity of real recursive functions. In: C. Calude, M. Dinneen, F. Peper (eds.) Unconventional Models of Computation, *Lecture Notes in Computer Science*, vol. 2509, pp. 1–14. Springer, Berlin (2002), third International Conference, UMC 2002, Kobe, Japan, October 15-19, 2002
103. Campagnolo, M.L.: Continuous time computation with restricted integration capabilities. Theoretical Computer Science **317**(4), 147–165 (2004)
104. Campagnolo, M.L., Moore, C., Costa, J.F.: Iteration, inequalities, and differentiability in analog computers. Journal of Complexity **16**(4), 642–660 (2001)
105. Campagnolo, M.L., Moore, C., Costa, J.F.: An analog characterization of the Grzegorczyk hierarchy. Journal of Complexity **18**(4), 977–1000 (2002)
106. Campagnolo, M.L., Ojakian, K.: The elementary computable functions over the real numbers: applying two new techniques. Archive for Mathematical Logic **46**(7–8), 593–627 (2008)
107. Cardelli, L.: Strand algebras for DNA computing. In: R. Deaton, A. Suyama (eds.) DNA Computing and Molecular Programming, *Lecture Note in Computer Science*, vol. 5877, pp. 12–24. Springer (2009)
108. Cardelli, L., Zavattaro, L.: Turing universality of the biochemical ground form. Mathematical Structures in Computer Science **20**(1), 45–73 (2010)
109. Care, C.: Technology for Modelling: Electrical Analogies, Engineering Practice, and the Development of Analogue Computing. Springer Science & Business Media (2010)
110. Carmantini, G.S., Graben, P.B., Desroches, M., Rodrigues, S.: Turing computation with recurrent artificial neural networks. In: Proceedings of the 2015 International Conference on Cognitive Computation: Integrating Neural and Symbolic Approaches, *CEUR Workshop Proceedings*, vol. 1583, pp. 1–9 (2015)
111. Cervelle, J.: Constructing continuous systems from discrete cellular automata. In: P. Bonizzoni, V. Brattka, B. Löwe (eds.) The Nature of Computation; Logic, Algorithms, Applications, *Lecture Notes in Computer Science*, vol. 7921, pp. 55–64. Springer (2013), 9th Conference on Computability in Europe, CiE 2013, Milan, Italy, July 2013
112. Chaitin, G.: The halting probability omega: Irreducible complexity in pure mathematics. Milan Journal of Mathematics **75**(1), 291–304 (2007)
113. Chatzigiannakis, I., Michail, O., Nikolaou, S., Pavlogiannis, A., Spirakis, P.: Passively mobile communicating machines that use restricted space. Theoretical Computer Science **412**, 6469–6483 (2011)
114. Chazelle, B.: Natural algorithms. In: Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 422–431. Society for Industrial and Applied Mathematics (2009)
115. Chazelle, B.: The convergence of bird flocking. Journal of the ACM (JACM) **61**(4), 21 (2014)
116. Chazelle, B.: An algorithmic approach to collective behavior. Journal of Statistical Physics **158**(3) (2015)
117. Chazelle, B.: Diffusive influence systems. SIAM Journal on Computing **44**(5), 1403–1442 (2015)
118. Chen, H.L., Doty, D., Soloveichik, D.: Deterministic function computation with chemical reaction networks. Natural Computing **13**(4), 517–534 (2014)

119. Chen, H.L., Doty, D., Soloveichik, D.: Rate-independent computation in continuous chemical reaction networks. In: Proceedings of the 5th Conference on Innovations in Theoretical Computer Science, pp. 313–326. ACM (2014)
120. Chen, Y.J., Dalchau, N., Srinivas, N., Phillips, A., Cardelli, L., Soloveichik, D., Seelig, G.: Programmable chemical controllers made from DNA. Nature Nanotechnology **8**(10), 755–762 (2013)
121. Chonev, V., Ouaknine, J., Worrell, J.: On the Skolem problem for continuous linear dynamical systems. arXiv preprint arXiv:1506.00695 (2015)
122. Chonev, V., Ouaknine, J., Worrell, J.: On recurrent reachability for continuous linear dynamical systems. In: Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, pp. 515–524. ACM (2016)
123. Church, G.M., Gao, Y., Kosuri, S.: Next-generation digital information storage in DNA. Science **337**(6102), 1628–1628 (2012)
124. Cockshott, P., Mackenzie, L., Michaelson, G.: Physical constraints on hypercomputation. Theoretical Computer Science **394**(3), 159–174 (2008)
125. Collins, P.: Continuity and computability on reachable sets. Theoretical Computer Science **341**, 162–195 (2005)
126. Collins, P., Lygeros, J.: Computability of finite-time reachable sets for hybrid systems. In: Proceedings of the 44th IEEE Conference on Decision and Control and the European Control Conference, pp. 4688–4693. IEEE Computer Society Press (2005)
127. Collins, P., van Schuppen, J.H.: Observability of hybrid systems and Turing machines. In: 43rd IEEE Conference on Decision and Control (2004)
128. Conrad, M.: Molecular computing. Advances in Computers **31**, 235–324 (1990)
129. Cook, M., Soloveichik, D., Winfree, E., Bruck, J.: Programmability of chemical reaction networks. In: A. Condon, D. Harel, J.N. Kok, A. Salomaa, E. Winfree (eds.) Algorithmic Bioprocesses, pp. 543–584. Springer (2009)
130. Copeland, B.J.: Even Turing machines can compute uncomputable functions. In: C.S. Calude, J. Casti, M.J. Dinneen (eds.) Unconventional Models of Computation. Springer-Verlag (1998)
131. Copeland, B.J.: Accelerating Turing machines. Minds and Machines **12**, 281–301 (2002)
132. Copeland, B.J.: Hypercomputation: philosophical issues. Theoretical Computer Science **317**(1–3), 251–267 (2004)
133. Copeland, B.J.: The Church-Turing thesis. In: E.N. Zalta (ed.) The Stanford Encyclopedia of Philosophy. Stanford University (Fall 2002)
134. Costa, J.F., Leong, R.: The ARNN model relativises P=NP and P!=NP. Theoretical Computer Science **499**, 2–22 (2013)
135. Costa, J.F., Loff, B., Mycka, J.: The new promise of analog computation. In: S.B. Cooper, B. Löwe, A. Sorbi (eds.) Computation and Logic in the Real World, *Lecture Notes in Computer Science*, vol. 4497, pp. 189–195. Springer, Berlin (2007), Third Conference on Computability in Europe, CiE 2007, Siena, Italy, June 18-23, 2007
136. Costa, J.F., Loff, B., Mycka, J.: A foundation for real recursive function theory. Annals of Pure and Applied Logic **160**(3), 255–288 (2009)
137. Costa, N., Doria, F.: Undecidable Hopf bifurcation with undecidable fixed point. International Journal of Theoretical Physics **33**(9), 1885–1903 (1994)
138. Costa, N., Doria, F., Amaral, A.: Dynamical system where proving chaos is equivalent to proving Fermat's conjecture. International Journal of Theoretical Physics **32**(11), 2187–2206 (1993)
139. Couturier, E., Jacquet, N.: Construction of a universal ordinary differential equation C^∞ of order 3. Bulletin des Sciences Mathématiques **150**, 62–83 (2019)
140. Cummings, R., Doty, D., Soloveichik, D.: Probability 1 computation with chemical reaction networks. In: S. Murata, S. Kobayashi (eds.) International Workshop on DNA-Based Computers, *Lecture Notes in Computer Science*, vol. 8727, pp. 37–52. Springer (2014)
141. Czyzowicz, J., Gasieniec, L., Kosowski, A., Kranakis, E., Spirakis, P.G., Uznański, P.: On convergence and threshold properties of discrete Lotka-Volterra population protocols. In:

M.M. Halldórsson, K. Iwama, N. Kobayashi, B. Speckmann (eds.) Automata, Languages, and Programming, *Lecture Notes in Computer Science*, vol. 9134, pp. 393–405. Springer, Berlin, Heidelberg (2015), 42nd International Colloquium, ICALP 2015, Kyoto, Japan, July 6–10, 1015, Proceedings, Part I
142. Davies, E.B.: Building infinite machines. The British Journal for the Philosophy of Science **52**, 671–682 (2001)
143. Davis, M.: The myth of hypercomputation. In: C. Teuscher (ed.) Alan Turing: Life and Legacy of a Great Thinker, pp. 195–211. Springer, Berlin, Heidelberg (2004)
144. Delporte-Gallet, C., Fauconnier, H., Guerraoui, R., Ruppert, E.: When birds die: Making population protocols fault-tolerant. In: P.B. Gibbons, T.F. Abdelzaher, J. Aspnes, R. Rao (eds.) Distributed Computing in Sensor Systems, Second IEEE International Conference, DCOSS 2006, San Francisco, CA, USA, June 18-20, 2006, Proceedings, *Lecture Notes in Computer Science*, vol. 4026, pp. 51–66. Springer (2006)
145. Delvenne, J.C., Kurka, P., Blondel, V.D.: Computational universality in symbolic dynamical systems. In: M. Margenstern (ed.) MCU: International Conference on Machines, Computations, and Universality, *Lecture Notes in Computer Science*, vol. 3354, pp. 104–115. Springer (2004)
146. Dershowitz, N., Gurevich, Y.: A natural axiomatization of computability and proof of Church's thesis. The Bulletin of Symbolic Logic **14**(3), 299–350 (2008)
147. Deutsch, D.: Quantum theory, the Church-Turing principle and the universal quantum computer. Proceedings of the Royal Society London A **400**, 97–117 (1985)
148. Diehl, P.U., Cook, M.: Unsupervised learning of digit recognition using spike-timing-dependent plasticity. Frontiers in Computational Neuroscience **9**, 99 (2015)
149. Doty, D.: Timing in chemical reaction networks. In: Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 772–784. Society for Industrial and Applied Mathematics (2014)
150. Dowek, G.: Les métamorphoses du calcul : une étonnante histoire de mathématiques. Le Pommier, Paris (2007)
151. Dowek, G.: The physical Church thesis as an explanation of the Galileo thesis. Natural Computing **11**(2), 247–251 (2012)
152. Durand-Lose, J.: Abstract geometrical computation: Turing-computing ability and undecidability. In: S.B. Cooper, B. Löwe, L. Torenvliet (eds.) New Computational Paradigms, First Conference on Computability in Europe, CiE 2005, Amsterdam, The Netherlands, June 8-12, 2005, Proceedings, *Lecture Notes in Computer Science*, vol. 3526, pp. 106–116. Springer (2005)
153. Durand-Lose, J.: Abstract geometrical computation 1: Embedding black hole computations with rational numbers. Fundam. Inform. **74**(4), 491–510 (2006)
154. Durand-Richard, M.J.: Charles Babbage (1791-1871) : de l'école algébrique anglaise à la 'machine analytique'. Mathématiques, Informatique et Sciences Humaines **118**, 5–31 (1992)
155. Durand-Richard, M.J.: Planimeters and integraphs in the 19th century. Before the differential analyser. Nuncius **25**(1), 101–124 (2010)
156. Durand-Richard, M.J.: Entre Ciel et Mer. Des observatoires pour l'enseignement de l'astronomie, des sciences maritimes et le service de l'heure, en France et en Europe, de la fin du XVIIIe au début du XXe siècle : institutions, pratiques et cultures, chap. De la prédiction des marées : entre calcul, observation et mécanisation (1831-1876). No. 8-9 in II (2016)
157. Durand-Richard, M.J.: De l'analyseur différentiel à l'ordinateur : le parcours de Douglas R. Hartree (1897-1958). Revue de Synthèse **139**(3–4), 289–326 (2017)
158. Earman, J., Norton, J.D.: Forever is a day: Supertasks in Pitowksy and Malament-Hogarth spacetimes. Philosophy of Science **60**(1), 22–42 (1993)
159. Ercsey-Ravasz, M., Toroczkai, Z.: Optimization hardness as transient chaos in an analog approach to constraint satisfaction. Nature Physics **7**(12), 966–970 (2011)
160. Esparza, J., Ganty, P., Leroux, J., Majumdar, R.: Verification of population protocols. Acta Informatica **54**, 191–215 (2016)

161. Etesi, G., Németi, I.: Non-Turing computations via Malament-Hogarth space-times. International Journal of Theoretical Physics **41**(2), 341–370 (2002)
162. Evans, C.G., Winfree, E.: Physical principles for DNA tile self-assembly. Chemical Society Reviews **46**(12), 3808–3829 (2017)
163. Fages, F.: Cells as machines: towards deciphering biochemical programs in the cell. In: R. Naturajan (ed.) International Conference on Distributed Computing and Internet Technology, *Lecture Notes in Computer Science*, vol. 8337, pp. 50–67. Springer (2014)
164. Fages, F., Le Guludec, G., Bournez, O., Pouly, A.: Strong Turing completeness of continuous chemical reaction networks and compilation of mixed analog-digital programs. In: J. Feret, H. Koeppl (eds.) Computational Methods in Systems Biology-CMSB 2017, *Lecture Notes in Computer Science*, vol. 10545, pp. 108–127. Springer (2017)
165. Fages, F., Soliman, S.: Abstract interpretation and types for systems biology. Theoretical Computer Science **403**(1), 52–70 (2008)
166. Faybusovich, L.: Dynamical systems which solve optimization problems with linear constraints. IMA Journal of Mathematical Control and Information **8**, 135–149 (1991)
167. Fischer, J.: Instrumente zur Mechanischen Integration. Ein Zwischenbericht. Brückenschläge **25**, 111–156 (1995)
168. Fischer, J.: Instrumente zur Mechanischen Integration II. Ein (weiterer) Zwischenbericht. Chemie-Kultur-Geschichte **65**, 143–155 (2002)
169. Franchette, F.: La thèse de l'hyper-calcul : enjeux et problèmes philosophiques. Philosophia Scientiæ **16**(3) (2012)
170. Francisco, A.P.L.: Finite automata over continuous time (2002), diploma Thesis. Universidade Técnica de Lisboa, Instituto Superior Técnico
171. Fredkin, E., Toffoli, T.: Conservative logic. In: A. Adamatzky (ed.) Collision-Based Computing, pp. 47–81. Springer (2002)
172. Friedman, H.: Algorithmic procedures, generalized Turing algorithms, and elementary recursion theory. In: R. Gandy, C. Yates (eds.) Logic Colloquium'69, *Studies in Logic and the Foundations of Mathematics*, vol. 61, pp. 361–389. North-Holland, Amsterdam (1971), proceedings of the Summer School and Colloquium in Mathematical Logic, Manchester, August 1969
173. Gandhi, A., Khoussainov, B., Liu, J.: Finite automata over structures. In: M. Agrawal, S.B. Cooper, A. Li (eds.) Theory and Applications of Models of Computation, *Lecture Notes in Computer Science*, vol. 7287, pp. 373–384. Springer, Berlin, Heidelberg (2012)
174. Gandy, R.: Church's thesis and principles for mechanisms. In: H.J. Barwise Jon amd Keisler, K. Kunen (eds.) The Kleene Symposium, *Studies in Logic and the Foundations of Mathematics*, vol. 101, pp. 123–148. North-Holland, Amsterdam-New York (1980), Proc. Sympos., Univ. Wisconsin, Madison, Wis., 1978
175. Gaßner, C.: The separation of relativized versions of P and DNP for the ring of the reals. J. UCS **16**(18), 2563–2568 (2010)
176. Gaßner, C.: Strong Turing degrees for additive BSS RAM's. Logical Methods in Computer Science **9**(4:25), 1–18 (2013)
177. Ghosh-Dastidar, S., Adeli, H.: Spiking neural networks. International Journal of Neural Systems **19**(04), 295–308 (2009)
178. Gill, P.E., Murray, W., Saunders, M.A., Tomlin, J.A., Wright, M.H.: On projected Newton barrier methods for linear programming and an equivalence to Karmarkar's projective method. Mathematical Programming **36**(2), 183–209 (1986)
179. Gillespie, D.: A rigorous derivation of the chemical master equation. Physica A **188**(1-3), 404–425 (1992)
180. Gillespie, D.T.: Exact stochastic simulation of coupled chemical reactions. J. Phys. Chem. **81**(25), 2340–2361 (1977)
181. Gladkikh, A., Malinetskii, G.: Study of dynamical systems from the viewpoint of complexity and computational capabilities. Differential Equations **52**(7), 897–905 (2016)
182. Gori, M., Meer, K.: A step towards a complexity theory for analog systems. Mathematical Logic Quarterly **48**(S1), 45–58 (2002)

183. Gori, M., Meer, K.: Some aspects of a complexity theory for continuous time systems. In: S.B. Cooper, B. Löwe, A. Sorbi (eds.) Computation and Logic in the Real World, *Lecture Notes in Computer Science*, vol. 4497, pp. 554–565. Springer (2007), third Conference on Computability in Europe, CiE 2007, Siena, Italy, June 18-23, 2007
184. Graça, D.S.: Some recent developments on Shannon's general purpose analog computer. Mathematical Logic Quarterly **50**(4,5), 473–485 (2004)
185. Graça, D.S., Campagnolo, M.L., Buescu, J.: Robust simulations of Turing machines with analytic maps and flows. In: S.B. Cooper, B. Löwe, L. Torenvliet (eds.) New Computational Paradigms, *Lecture Notes in Computer Science*, vol. 3526, pp. 169–179. Springer, Berlin (2005), first Conference on Computability in Europe, CiE 2005, Amsterdam, The Netherlands, June 2005
186. Graça, D.S., Costa, J.F.: Analog computers and recursive functions over the reals. Journal of Complexity **19**(5), 644–664 (2003)
187. Graves, A., Wayne, G., Danihelka, I.: Neural Turing machines. arXiv preprint arXiv:1410.5401 (2014)
188. Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwińska, A., Colmenarejo, S.G., Grefenstette, E., Ramalho, T., Agapiou, J., et al.: Hybrid computing using a neural network with dynamic external memory. Nature **538**(7626), 471–476 (2016)
189. Grefenstette, E., Hermann, K.M., Suleyman, M., Blunsom, P.: Learning to transduce with unbounded memory. In: Advances in Neural Information Processing Systems, pp. 1828–1836 (2015)
190. Gruska, J.: Foundations of Computing. International Thomson Publishing (1997)
191. Gu, J., Gu, Q., Du, D.: On optimizing the satisfiability (SAT) problem. Journal of Computer Science and Technology **14**(1), 1–17 (1999)
192. Guerraoui, R., Ruppert, E.: Names trump malice: Tiny mobile agents can tolerate byzantine failures. In: S. Albers, A. Marchetti-Spaccamela, Y. Matias, N. Nikoletseas, W. Thomas (eds.) Automata, Languages and Programming, *Lecture Notes in Computer Science*, vol. 5556, pp. 484–495. Springer (2009)
193. Gupta, I., Nagda, M., Devaraj, C.F.: The design of novel distributed protocols from differential equations. Distributed Computing **20**(2), 95–114 (2007)
194. Gurevich, Y.: Sequential abstract-state machines capture sequential algorithms. ACM Transactions on Computational Logic **1**(1), 77–111 (2000)
195. Haigh, T., Priestley, P.M., , Rope, C.: ENIAC in Action: Making and Remaking the Modern Computer. MIT Press (2016)
196. Hartmann, L., Jones, N.D., Simonsen, J.G.: Programming in biomolecular computation. Electronic Notes in Theoretical Computer Science **268**, 97–114 (2010)
197. Hashagen, U.: Rechner für die Wissenschaft: "Scientific Computing" und Informatik im deutschen Wissenschaftssystem 1870-1970. Rechnende Maschinen im Wandel: Mathematik, Technik, Gesellschaft. Deutsches Museum pp. 111–152 (2011)
198. Hashagen, U.: Analog computing as a failed modernization program in Germany 1930-1960 (2017), talk at "Histoire et Philosophie de l'Informatique", MESHS, Lille
199. Head, T.: Formal language theory and DNA: An analysis of the generative capacity of specific recombinant behaviors. Bulletin of Mathematical Biology **49**, 737–759 (1987)
200. Helmke, U., Moore, J.B.: Optimization and Dynamical Systems. *Communications and Control Engineering Series*. Springer Verlag, London (1994)
201. Helmke, U., Moore, J.B.: Optimization and dynamical systems. Springer Science & Business Media (2012)
202. Henzinger, T.A., Kopke, P.W., Puri, A., Varaiya, P.: What's decidable about hybrid automata? Journal of Computer and System Sciences **57**(1), 94–124 (1998)
203. Hiebeler, D.: Dynamics and resistance to neighborhood perturbations of discrete- and continuous-time cellular automata. J. Cellular Automata **1**(2), 125–139 (2006)
204. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
205. Hofbauer, J., Sigmund, K.: Evolutionary game dynamics. Bulletin of the American Mathematical Society **4**, 479–519 (2003)

206. Hogarth, M.: Non-Turing computers and non-Turing computability. In: Proceedings of the Philosophy of Science Association (PSA'94), vol. 1, pp. 126–138 (1994)
207. Hogarth, M.: Predictability, computability and spacetime. Ph.D. thesis. Sidney Sussex College, Cambridge University (1996)
208. Hogarth, M.: Non-Turing computers are the new non-Euclidean geometries. Unconventional Computing **5**(3–4), 277–291 (2009)
209. Hogarth, M.L.: Does general relativity allow an observer to view an eternity in a finite time? Foundations of Physics Letters **5**, 173–181 (1992)
210. Hopfield, J.J.: Neural networks with graded responses have collective computational properties like those of two-state neurons. Proceedings of the National Academy of Sciences of the United States of America **81**, 3088–3092 (1984)
211. Huang, X., Klinge, T.H., Lathrop, J.I., Li, X., Lutz, J.H.: Real-time computability of real numbers by chemical reaction networks. In: M.J. Patitz, M. Stannett (eds.) International Conference on Unconventional Computation and Natural Computation, *Lecture Notes in Computer Science*, vol. 10240, pp. 29–40. Springer (2017)
212. Jeandel, E.: Topological automata. Theory of Computing Systems **40**(4), 397–407 (2007)
213. Karmarkar, N.: A new polynomial-time algorithm for linear programming. In: Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing, pp. 302–311. ACM (1984)
214. Kempe, A.: On a general method of describing plane curves of the n–th degree by linkwork. Proceedings of the London Mathematical Society **7**, 213–216 (1876)
215. Khachiyan, L.G.: Polynomial algorithms in linear programming. USSR Computational Mathematics and Mathematical Physics **20**(1), 53–72 (1980)
216. Kieu, T.D.: Hypercomputation with quantum adiabatic processes. Theoretical Computer Science **317**(1-3), 93–104 (2004)
217. Koiran, P.: The topological entropy of iterated piecewise affine maps is uncomputable. Discrete Mathematics & Theoretical Computer Science **4**(2), 351–356 (2001)
218. Koiran, P., Cosnard, M., Garzon, M.: Computability with low-dimensional dynamical systems. Theoretical Computer Science **132**(1-2), 113–128 (1994)
219. Koiran, P., Moore, C.: Closed-form analytic maps in one and two dimensions can simulate universal Turing machines. Theoretical Computer Science **219**, 217–223 (1999)
220. Krivine, H., Lesne, A., Treiner, J.: Discrete-time and continuous-time modelling: some bridges and gaps. Mathematical Structures in Computer Science **17**(2), 261–276 (2007)
221. Kurganskyy, O., Potapov, I.: Computation in one-dimensional piecewise maps and planar pseudo-billiard systems. In: C. Calude, M.J. Dinneen, G. Paun, M.J. Pérez-Jiménez, G. Rozenberg (eds.) Unconventional Computation, 4th International Conference, UC 2005, Sevilla, Spain, October 3-7, 2005, Proceedings, *Lecture Notes in Computer Science*, vol. 3699, pp. 169–175. Springer (2005)
222. Kurtz, T.G.: Approximation of Population Processes, *CBMS-NSF Regional Conference Series in Applied Mathematics*, vol. 36. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, Pa. (1981)
223. Lanotte, R., Tini, S.: Taylor approximation for hybrid systems. Information and Computation **205**(11), 1575–1607 (2007)
224. Lee, E.A.: Models of time for CPS. Course at UC Berkeley (available online) (2015)
225. Legenstein, R., Maass, W.: What makes a dynamical system computationally powerful? In: S. Haykin, J.C. Principe, T. Sejnowski, J. McWhirter (eds.) New Directions in Statistical Signal Processing: From Systems to Brain, pp. 127–154. MIT Press (2007)
226. Lesne, A.: The discrete versus continuous controversy in physics. Mathematical Structures in Computer Science **17**(2), 185–223 (2007)
227. Lipshitz, L., Rubel, L.A.: A differentially algebraic replacement theorem, and analog computability. Proceedings of the American Mathematical Society **99**(2), 367–372 (1987)
228. Lipton, R.J.: DNA solution of hard computational problems. Science **268**, 542–545 (1995)
229. Loff, B.: A functional characterisation of the analytical hierarchy. In: Computability in Europe 2007: Computation and Logic in the Real World. (2007), https://www.math.tecnico.ulisboa.pt/∼bloff/files/ranalytical.pdf

230. Loff, B., Costa, J.F., Mycka, J.: Computability on reals, infinite limits and differential equations. Applied Mathematics and Computation **191**(2), 353–391 (2007)
231. Longo, G., Paul, T.: The mathematics of computing between logic and physics. In: S.B. Cooper, A. Sorbi (eds.) Computability in context, pp. 243–273. Imperial College Press, London (2011)
232. Lovgren, S.: Computer made from DNA and enzymes. National Geographic News **24** (2003)
233. Maass, W., Bishop, C.: Pulsed Neural Networks. MIT Press, Cambridge MA. (1998)
234. Maass, W., Orponen, P.: On the effect of analog noise in discrete-time analog computations. Neural Computation **10**(5), 1071–1095 (1998)
235. Maass, W., Sontag, E.: Analog neural nets with gaussian or other common noise distributions cannot recognize arbitrary regular languages. Neural Computation **11**(3), 771–782 (1999)
236. MacLennan, B.: Field computation: A theoretical framework for massively parallel analog computation. parts i-iv. Technical Report CS-90-100, University of Tennessee, Computer Science Department (1990)
237. MacLennan, B.: The promise of analog computation. International Journal of General Systems **43**(7), 682–696 (2014)
238. MacLennan, B.J.: Field computation in natural and artificial intelligence. Information Sciences **119**(1), 73–89 (1999)
239. MacLennan, B.J.: Natural computation and non-Turing models of computation. Theoretical computer science **317**(1), 115–145 (2004)
240. MacLennan, B.J.: Analog computation. In: R.A. Meyers (ed.) Encyclopedia of Complexity and Systems Science, pp. 271–294. Springer (2009)
241. Margolus, N.: Physics-like models of computation. Physica D: Nonlinear Phenomena **10**(1-2), 81–95 (1984)
242. Mather, J., McGehee, R.: Solutions of the collinear four body problem which become unbounded in finite time. In: J. Moser (ed.) Dynamical Systems, Theory and Applications, *Lecture Notes in Physics*, vol. 38, pp. 573–597. Springer (1975)
243. Meer, K.: On Ladner's result for a class of real machines with restricted use of constants. Information and Computation **210**, 13–20 (2012)
244. Meer, K., Naif, A.: Generalized finite automata over real and complex numbers. Theoretical Computer Science **591**, 85–98 (2015)
245. Meer, K., Naif, A.: Periodic generalized automata over the reals. In: A.H. Dedin, J. Janousek, C. Martin-Vide, B. Truthe (eds.) International Conference on Language and Automata Theory and Applications, *Lecture Notes in Computer Science*, vol. 9618, pp. 168–180. Springer (2016)
246. Merolla, P.A., Arthur, J.V., Alvarez-Icaza, R., Cassidy, A.S., Sawada, J., Akopyan, F., Jackson, B.L., Imam, N., Guo, C., Nakamura, Y., et al.: A million spiking-neuron integrated circuit with a scalable communication network and interface. Science **345**(6197), 668–673 (2014)
247. Messick, S.: Compactness in the theory of continuous automata. In: S. Artemov, A. Nerode (eds.) International Symposium on Logical Foundations of Computer Science, *Lecture Notes in Computer Science*, vol. 9537, pp. 251–266. Springer (2016)
248. Michail, O., Chatzigiannakis, I., Spirakis, P.G.: New models for population protocols. Synthesis Lectures on Distributed Computing Theory **2**(1), 1–156 (2011)
249. Mickelson, N.B.: Analog computing device (1963), US Patent 3,113,170
250. Milici, P.: Tractional motion machines extend GPAC-generable functions. IJUC **8**(3), 221–233 (2012)
251. Milici, P.: Tractional motion machines: Tangent-managing planar mechanisms as analog computers and educational artifacts. In: J. Durand-Lose, N. Jonoska (eds.) Unconventional Computation and Natural Computation, *Lecture Notes in Computer Science*, vol. 7445, pp. 164–173. Springer (2012), 11th International Conference, UCNC 2012, Orléan, France, September 3-7, 2012. Proceedings
252. Mills, J.: Programmable VLSI extended analog computer for cyclotron beam control. Tech. Rep. 441, Indiana University Computer Science (1995)

253. Mills, J.W.: The nature of the extended analog computer. Physica D: Nonlinear Phenomena **237**(9), 1235 – 1256 (2008)
254. Mills, J.W., Himebaugh, B., Allred, A., Bulwinkle, D., Deckard, N., Gopalakrishnan, N., Miller, J., Miller, T., Nagai, K., Nakamura, J., Ololoweye, B., Vlas, R., Whitener, P., Ye, M., , Zhang, C.: Extended analog computers: A unifying paradigm for VLSI, plastic and colloidal computing systems. In: Workshop on Unique Chips and Systems (UCAS-1). Held in conjunction with IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS05). Austin, Texas (2005)
255. Moore, C.: Unpredictability and undecidability in dynamical systems. Physical Review Letters **64**(20), 2354–2357 (1990)
256. Moore, C.: Recursion theory on the reals and continuous-time computation. Theoretical Computer Science **162**(1), 23–44 (1996)
257. Moore, C., Mertens, S.: The Nature of Computation. OUP Oxford (2011)
258. Moschovakis, Y.N.: Abstract recursion as a foundation for the theory of algorithms. In: E. Börger, W. Oberschelp, M.M. Richter, B. Schinzel, W. Thomas (eds.) Computation and Proof Theory, *Lecture Notes in Mathematics*, vol. 1104, pp. 289–364. Springer-Verlag, Berlin (1984)
259. Mycka, J., Coelho, F., Costa, J.F.: The Euclid abstract machine: Trisection of the angle and the halting problem. In: C.S. Calude, M.J. Dinneen, G. Paun, G. Rozenberg, S. Stepney (eds.) Unconventional Computation, 5th International Conference, UC 2006, York, UK, September 4-8, 2006, Proceedings, *Lecture Notes in Computer Science*, vol. 4135, pp. 195–206. Springer (2006)
260. Mycka, J., Costa, J.F.: Real recursive functions and their hierarchy. Journal of Complexity **20**(6), 835–857 (2004)
261. Mycka, J., Costa, J.F.: What lies beyond the mountains? Computational systems beyond the Turing limit. European Association for Theoretical Computer Science Bulletin **85**, 181–189 (2005)
262. Mycka, J., Costa, J.F.: The $P \neq NP$ conjecture in the context of real and complex analysis. Journal of Complexity **22**(2), 287–303 (2006)
263. Nagamatu, M., Yanaru, T.: On the stability of Lagrange programming neural networks for satisfiability problems of prepositional calculus. Neurocomputing **13**(2), 119–133 (1996)
264. Neher, M.: Interval methods and Taylor model methods for ODEs. TM VII, Key West, KIT, Michigan State University p. 33 (2011)
265. Németi, I., Andréka, H.: New physics and hypercomputation. In: J. Wiedermann, G. Tel, J. Pokorný, M. Bieliková, J. Stuller (eds.) SOFSEM 2006: Theory and Practice of Computer Science, 32nd Conference on Current Trends in Theory and Practice of Computer Science, Merín, Czech Republic, January 21-27, 2006, Proceedings, *Lecture Notes in Computer Science*, vol. 3831, p. 63. Springer (2006)
266. Németi, I., Gyula, D.: Relativistic computers and the Turing barrier. Applied Mathematics and Computation **178**(1), 118–142 (2006)
267. Orponen, P.: A survey of continuous-time computation theory. In: D.Z. Du, K.I. Ko (eds.) Advances in Algorithms, Languages, and Complexity, pp. 209–224. Kluwer Academic Publishers (1997)
268. Orponen, P., Matamala, M.: Universal computation by finite two-dimensional coupled map lattices. In: Proceedings, Physics and Computation, pp. 243–7. New England Complex Systems Institute Cambridge, MA (1996)
269. Owens, L.: Vannevar Bush and the differential analyzer: the text and context of an early computer. Technology and Culture **27**(1), 63–95 (1986)
270. Păun, G.: Membrane Computing. An Introduction. Springer-Verlag, Berlin (2002)
271. Paun, G., Rozenberg, G.: A guide to membrane computing. Theoretical Computer Science **287**(1), 73–100 (2002)
272. Pégny, M.: Les deux formes de la thèse de Church-Turing et l'épistémologie du calcul. Philosophia Scientiæ **16**(3) (2012)

273. Pégny, M.: Computational complexity: An empirical view. In: 39th Annual Convention of the Society for the Study of Artificial Intelligence and the Simulation of Behaviour (AISB 2013), *Proceedings of the 2013 AISB Convention*. Exeter, United Kingdom (2013)
274. Pégny, M.: Sur les limites empiriques du calcul: calculabilité, complexité et physique. Ph.D. thesis. Paris 1 (2013)
275. Pégny, M.: How to make a meaningful comparison of models: The Church–Turing thesis over the reals. Minds and Machines **26**(4), 359–388 (2016)
276. Platzer, A.: Differential dynamic logic for hybrid systems. J. Autom. Reas. **41**(2), 143–189 (2008)
277. Platzer, A.: Logical Analysis of Hybrid Systems: Proving Theorems for Complex Dynamics. Springer Science & Business Media (2010)
278. Platzer, A.: The complete proof theory of hybrid systems. In: Proceedings of the 2012 27th Annual IEEE/ACM Symposium on Logic in Computer Science, pp. 541–550. IEEE Computer Society (2012)
279. Platzer, A.: A uniform substitution calculus for differential dynamic logic. In: A.P. Felty, A. Middeldorp (eds.) International Conference on Automated Deduction, *Lecture Notes in Computer Science*, vol. 9195, pp. 467–481. Springer (2015)
280. Platzer, A.: Logic & proofs for cyber-physical systems. In: N. Olivetti, A. Tiwari (eds.) International Joint Conference on Automated Reasoning, *Lecture Notes in Computer Science*, vol. 9706, pp. 15–21. Springer (2016)
281. Platzer, A.: Logical Foundations of Cyber-Physical Systems. Springer, Switzerland (2018)
282. Poças, D., Zucker, J.: Fixed point techniques in analog systems. In: J. Bélair, I.A. Frigaard, H. Kunze, R. Makarov, R. Melnik, R.J. Spiteri (eds.) Mathematical and Computational Approaches in Advancing Modern Science and Engineering, pp. 701–711. Springer, Cham (2016)
283. Poizat, B.: Les Petits Cailloux: Une Approche Modèle-Théorique de l'Algorithmie. Aléas Editeur (1995)
284. Pour-El, M.B.: Abstract computability and its relation to the general purpose analog computer (some connections between logic, differential equations and analog computers). Transactions of the American Mathematical Society **199**, 1–28 (1974)
285. Qian, L., Soloveichik, D., Winfree, E.: Efficient Turing-universal computation with DNA polymers. In: Y. Sakikabara, Y. Mi (eds.) Proc. DNA Computing and Molecular Programming, *LNCS*, vol. 6518, pp. 123–140. Springer-Verlag (2011)
286. Rabinovich, A.: Automata over continuous time. Theoretical Computer Science **300**(1–3), 331–363 (2003)
287. Rabinovich, A.M., Trakhtenbrot, B.A.: From finite automata toward hybrid systems (extended abstract). In: B.S. Chlebus, L. Czaja (eds.) FCT, *Lecture Notes in Computer Science*, vol. 1279, pp. 411–422. Springer (1997)
288. Renegar, J.: A Mathematical View of Interior-Point Methods in Convex Optimization. SIAM (2001)
289. Reus, B.: Limits of Computation: From a Programming Perspective, chap. Molecular Computing. Springer (2016)
290. Rojas, R., Hashagen, U.: The First Computers: History and Architectures. MIT Press (2002)
291. Rozenberg, G., Bäck, T., Kok, J.N.: Handbook of Natural Computing. Springer (2011)
292. Rubel, L.A.: A universal differential equation. Bulletin of the American Mathematical Society **4**(3), 345–349 (1981)
293. Rubel, L.A.: The extended analog computer. Advances in Applied Mathematics **14**, 39–50 (1993)
294. Ruohonen, K.: Undecidability of event detection for ODEs. Journal of Information Processing and Cybernetics **29**, 101–113 (1993)
295. Ruohonen, K.: Event detection for ODEs and nonrecursive hierarchies. In: J. Karhumäki, H. Maurer, G. Rozenberg (eds.) Results and trends in theoretical computer science (Graz, 1994), *Lecture Notes in Computer Science*, vol. 812, pp. 358–371. Springer, Berlin (1994)
296. Ruohonen, K.: Undecidable event detection problems for ODEs of dimension one and two. RAIRO Informatique Théorique et Applications **31**(1), 67–79 (1997)

297. Ruohonen, K.: Chomskian hierarchies of families of sets of piecewise continuous functions. Theory of Computing Systems **37**(5), 609–638 (2004)
298. Saari, D.G., Xia, Z.: Off to infinity in finite time. Notices of the AMS **42**(5) (1995)
299. Scalise, D., Schulman, R.: Designing modular reaction-diffusion programs for complex pattern formation. Technology **2**(01), 55–66 (2014)
300. Scalise, D., Schulman, R.: Emulating cellular automata in chemical reaction–diffusion networks. Natural Computing **15**(2), 197–214 (2016)
301. Schaller, M., Svozil, K.: Scale-invariant cellular automata and self-similar Petri nets. The European Physical Journal B **69**(2), 297–311 (2009)
302. Schönhage, A.: Storage modification machines. SIAM Journal on Computing **9**(3), 490–508 (1980)
303. Shannon, C.E.: Mathematical theory of the differential analyser. Journal of Mathematics and Physics MIT **20**, 337–354 (1941)
304. Shor, P.W.: Algorithms for quantum computation: Discrete logarithms and factoring. In: S. Goldwasser (ed.) Proceedings of the 35th Annual Symposium on Foundations of Computer Science, pp. 124–134. IEEE Computer Society Press, Los Alamitos, CA, USA (1994)
305. Siegelmann, H.T.: Neural Networks and Analog Computation: Beyond the Turing Limit. Birkhäuser (1999)
306. Siegelmann, H.T., Sontag, E.D.: Analog computation via neural networks. Theoretical Computer Science **131**(2), 331–360 (1994)
307. Siegelmann, H.T., Sontag, E.D.: On the computational power of neural nets. Journal of Computer and System Sciences **50**(1), 132–150 (1995)
308. Šíma, J., Orponen, P.: Continuous-time symmetric Hopfield nets are computationally universal. Neural Computation **15**(3), 693–733 (2003)
309. Šíma, J., Orponen, P.: General-purpose computation with neural networks: A survey of complexity theoretic results. Neural Computation **15**(12), 2727–2778 (2003)
310. Smith, W.D.: Plane mechanisms and the downhill principle (1998)
311. Smith, W.D.: Church's thesis meets quantum mechanics. Tech. rep., NEC Research Institute (2004)
312. Smith, W.D.: Church's thesis meets the N-body problem. Applied Mathematics and Computation **178**(1), 154–183 (2006)
313. Soloveichik, D.: The computational power of chemical reaction networks. Banff International Research Station. (2014)
314. Soloveichik, D., Cook, M., Winfree, E., Bruck, J.: Computation with finite stochastic chemical reaction networks. Natural Computing **7**(4), 615–633 (2008)
315. Soloveichik, D., Seelig, G., Winfree, E.: DNA as a universal substrate for chemical kinetics. Proceedings of the National Academy of Sciences **107**(12), 5393–5398 (2010)
316. Soloveichik, D., Winfree, E.: Complexity of self-assembled shapes. SIAM Journal on Computing **36**(6), 1544–1569 (2007)
317. Stemmer, W.P.: The evolution of molecular computation. Science **270**(5241), 1510–1511 (1995)
318. Sun, G., Giles, C., Chen, H., Lee, Y.: The neural network pushdown automaton: model, stack and learning simulations. arXiv preprint arXiv:1711.05738 (1993)
319. Syropoulos, A.: Hypercomputation: Computing Beyond the Church-Turing Barrier. Springer Science & Business Media (2008)
320. Tero, A., Kobayashi, R., Nakagaki, T.: A mathematical model for adaptive transport network in path finding by true slime mold. Journal of Theoretical Biology **244**(4), 553–564 (2007)
321. Thomson, W.: On an instrument for calculating the integral of the product of two given functions. In: Proceedings of the Royal Society of London, vol. 24, pp. 266–276 (1876)
322. Trakhtenbrot, B.A.: Automata and their interaction: Definitional suggestions. In: G. Ciobanu, G. Paun (eds.) FCT, *Lecture Notes in Computer Science*, vol. 1684, pp. 54–89. Springer (1999)
323. Tucker, J., Zucker, J.: Computable functions and semicomputable sets on many-sorted algebras. In: S. Abramsky, D. Gabbay, T. Maibaum (eds.) Handbook of Logic in Computer Science, Volume 5, pp. 317–523. Oxford University Press, Oxford (2000)

324. Tucker, J.V., Zucker, J.I.: Computability of analog networks. Theoretical Computer Science **371**(1–2), 115–146 (2007)
325. Tucker, J.V., Zucker, J.I.: Computability of operators on continuous and discrete time streams. Computability **3**(1), 9–44 (2014)
326. Ulmann, B.: Analog Computing. Walter de Gruyter (2013)
327. Vergis, A., Steiglitz, K., Dickinson, B.: The complexity of analog computation. Mathematics and Computers in Simulation **28**(2), 91–113 (1986)
328. Wah, B.W., Chang, Y.J.: Trace-based methods for solving nonlinear global optimization and satisfiability problems. Journal of Global Optimization **10**(2), 107–141 (1997)
329. Weibull, J.W.: Evolutionary Game Theory. The MIT Press (1995)
330. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)
331. Welch, P.: The extent of computation in Malament-Hogarth spacetimes. British Journal for the Philosophy of Science **59**(4), 659–674 (2008)
332. Whyman, R.: Physical computation, $P/poly$ and $P/log\star$. In: A.A. Abbott, D.C. Horsman (eds.) Proceedings of the 7th International Workshop on Physics and Computation, *Electronic Proceedings in Theoretical Computer Science*, vol. 214, pp. 41–52. EPTCS (2016)
333. Williams, M.R.: A History of Computing Technology, 2nd edn. Wiley-IEEE Computer Society Press (1997)
334. Woods, D., Naughton, T.J.: An optical model of computation. Theoretical Computer Science **334**(1-3), 227–258 (2005)
335. Yin, X., Sedighi, B., Varga, M., Ercsey-Ravasz, M., Toroczkai, Z., Hu, X.S.: Efficient analog circuits for boolean satisfiability. arXiv preprint arXiv:1606.07467 (2016)

Chapter 7
Computable Measure Theory and Algorithmic Randomness

Mathieu Hoyrup and Jason Rute

Abstract We provide a survey of recent results in computable measure and probability theory, from the perspectives of both computable analysis and algorithmic randomness, and discuss the relations between them.

7.1 Introduction

The underlying topic of this chapter is computable probability theory, considered from two angles. The first angle follows the traditional approach of computable analysis, where the goal is to develop effective versions of classical notions and to study the effectiveness of classical theorems. These notions and theorems mainly come from measure theory, in which probability theory is grounded. Therefore, the first part of this chapter is devoted to computable measure theory. The second direction is the algorithmic theory of randomness, whose goal was originally to define what it means for an individual object to be random, using computability theory. These two approaches have been developed in parallel for a long time, but their interaction has recently become a fruitful research direction, surveyed in this chapter.

Let us give a quick overview of the typical questions that are studied in the vast area of computable probability theory. A recurrent topic in computable analysis is to investigate the effectiveness of existence theorems. Many theorems in probability theory are convergence theorems. How to analyze such theorems from a computability perspective? A theorem stating the convergence of a sequence can be presented as an existence theorem: it asserts the existence of an index from which the terms

Mathieu Hoyrup
Université de Lorraine, CNRS, Inria, LORIA, Nancy, France
e-mail: mathieu.hoyrup@inria.fr

Jason Rute
Department of Mathematics, Pennsylvania State University, State College, Pennsylvania, USA
e-mail: jason.rute@uwalumni.com

V. Brattka, P. Hertling (eds.), *Handbook of Computability and Complexity in Analysis*,
Theory and Applications of Computability, https://doi.org/10.1007/978-3-030-59234-9_7

of the sequence are close to the limit. This leads to the investigation of the computability of the speed of convergence, i.e. of the aforementioned index given the prescribed distance to the limit. It happens that many convergence theorems are not computable in this way (for instance, martingale convergence theorems or ergodic theorems). This was observed by Bishop in the context of constructive analysis, in his *Foundations of Constructive Analysis* (p. 214 in [19]):

> Certain parts of measure theory are hard to develop constructively, because limits that are classically proved to exist simply do not exist constructively.

So it looks like it is the end of the story. However, there is another way of interpreting an almost sure convergence theorem as an existence theorem: it states the existence of a set of full measure on which pointwise convergence holds. So the computability problem amounts to studying the computability of this full-measure set.

Some of the computable approaches to measure theory fail to provide useful computability notions for full-measure sets: the full-measure sets are all computable because they are all equivalent to the whole space. One needs a finer look into the computability of those sets, and Algorithmic Randomness provides this.

Algorithmic Randomness starts with Martin-Löf's seminal paper [85], where he introduces a notion of effective null set, nowadays called a *Martin-Löf null set*. Such a notion enables one to define what a random point is: it is a point that does not belong to any effective null set, which makes sense as there are countably many such sets hence their union is again a null set. Since then Algorithmic Randomness has been studied in several directions. We will only present the part that interacts with computable measure and probability theory and will not mention its interactions with computability theory and Kolmogorov complexity, a large part of which can be found in [82, 95, 32].

The goals of this chapter are to present the main results obtained in recent years, a comprehensive bibliography as well as the basic definitions, tools and techniques needed for this development. We include proofs of simple results, some of them appearing nowhere explicitly, so that the reader can quickly understand how they work and confidently use them. As for the deeper and more complicated results, we refer to the corresponding research articles, where the proofs can be found. When possible, we give an outline of the proof, or at least some intuition about how it works.

7.2 Computable Measure Theory

7.2.1 Background from Computable Analysis

We assume familiarity with basic notions from computability theory (computable set of natural numbers, computably enumerable (c.e.) set, computable function, etc.). We give a couple of central notions from computable analysis. References

will be given where needed. The standard reference for computability on countably based spaces is [116].

A *computable metric space* is a triple (X,d,S) where (X,d) is a separable metric space and $S=(s_i)_{i\in\mathbb{N}}$ is a sequence of points of X, called *simple* points, such that the reals numbers $d(s_i,s_j)$ are uniformly computable. A *name* of a point $x\in X$ is a function $f:\mathbb{N}\to\mathbb{N}$ such that $d(x,s_{f(i)})<2^{-i}$ for all i. A point is *computable* if it has a computable name. The basic metric balls are the metric balls centered at simple points with rational radii, and have a canonical indexing $(B_i)_{i\in\mathbb{N}}$. An open set $U\subseteq X$ is an *effectively open set* if $U=\bigcup_{i\in E}B_i$ for some c.e. set $E\subseteq\mathbb{N}$. A function $f:X\to Y$ between computable metric spaces is computable if there is an oracle Turing machine reading a name of x and outputting a name of $f(x)$. Equivalently, f is computable if for each basic ball $B\subseteq Y$ the pre-image $f^{-1}(B)$ is an effectively open subset of X, uniformly in the index of B.

A compact set $K\subseteq X$ is an *effectively compact set* if there is a computable enumeration of the finite sets $F\subseteq\mathbb{N}$ such that $K\subseteq\bigcup_{i\in F}B_i$. Let $K\subseteq X$ be an effectively compact set. Its complement $X\setminus K$ is effectively open. If U is effectively open then $K\setminus U$ is effectively compact. If $f:K\to Y$ is computable then $f(K)$ is effectively compact. If, moreover, f is one-to-one then $f^{-1}:f(K)\to K$ is computable.

For instance, $\mathbb{R}$ with the Euclidean metric and a canonical enumeration $(q_i)_{i\in\mathbb{N}}$ of the rational numbers is a computable metric space. We denote by $\mathbb{R}_<$ the set of real numbers with a different naming system, or representation. In that space, a name of $x\in\mathbb{R}_<$ is a function $f:\mathbb{N}\to\mathbb{N}$ such that $x=\sup_i q_{f(i)}$. A real number is *lower semicomputable* or *left-c.e.* if it has a computable name in this sense.

7.2.2 Framework

The most general way of defining measures goes through the abstract notions of ring, algebra, σ-ring, σ-algebra, outer measure, etc. Effective counterparts to this development have been investigated and used in several articles [120, 121, 69, 70, 119, 118].

Another approach, less general but covering a wide range of applications, is to restrict oneself to spaces with a structure (for instance a separable metric) on which computable analysis is already settled, and to work with the measurable structure induced by it (for instance the Borel σ-algebra generated by the open sets). Many articles in the literature follow this approach, which we will adopt here for the following reasons.

The practical reason is that manipulating a computable measurable structure, in addition to other structures, is rather heavy. The mathematical reason is that results from ordinary measure and probability theory are often stated on spaces with more structure than just a measurable structure, for instance a metric, and hold only on such spaces. As a result, the computable structures from mainstream computable analysis are usually sufficient to investigate computable measure theory. Finally, the approach of working with a fixed ring shows its limits when the measure is not fixed

and computable but is a variable object of the problem. This limitation manifests itself in several ways. Each ring induces its own notion of computable measure, even among the computable rings, so the underlying ring should evolve at the same time as the measure. The topology induced by the representation of measures associated with a fixed ring is not the important weak* topology. Intuitively, fixing a ring and representing a measure by giving the weights of the ring elements introduces artificial discontinuities, similarly to representing real numbers by their binary expansions.

For all these reasons we have chosen to follow the second approach by working on computable metric spaces only and considering Borel measures. This avoids too many additional definitions and it is sufficient for most purposes. Moreover we will restrict our attention to probability measures for simplicity.

We are implicitly working in the framework of Type-Two computability and represented spaces, but there are other options. A complete approach to computable measure theory has been developed by Edalat [33, 34] in the framework of domain theory.

7.2.2.1 Representing and Computing with Probability Measures

Let X be a computable metric space. A Borel probability measure can be equivalently represented as (see [49, 107] for instance) :

- A function $\mathscr{O}(X) \to \mathbb{R}_<$ mapping an open set U to $\mu(U)$,
- A function $\mathscr{C}(X,[0,1]) \to \mathbb{R}$ mapping a bounded continuous function f to $\int f \, \mathrm{d}\mu$,
- A function $\mathbb{N} \to \mathbb{R}_<$ mapping an index of a finite union of balls to its weight,
- A point in the computable metric space $\mathscr{M}_1(X)$ of Borel probability measures endowed with the Prokhorov metric.

Here $\mathbb{R}$ denotes the space of real numbers with the Euclidean topology while $\mathbb{R}_<$ denotes the space of real numbers with the topology induced by the semi-lines $(x, +\infty)$.

In particular, we have the following.

Definition 7.2.1 (Computable measure). A Borel probability measure μ is *computable* if the following equivalent conditions hold:

- The measure of a finite union of basic metric balls is uniformly lower semicomputable,
- The measure of an effectively open set is uniformly lower semicomputable,
- The integral of a bounded computable function $h : X \to [0,1]$ is uniformly computable.

More generally, a finite measure μ is computable if $\mu(X)$ is a computable real number and the normalized measure $\mu/\mu(X)$ is a computable probability measure. Computability of σ-finite or general measures can be defined in similar ways, with variations.

If μ is a computable Borel probability measure then the measure of a basic ball $B(s,r)$ is lower semicomputable but is not necessarily computable. However,

one can take a class of radii other than the rationals to make the measures of basic balls computable. This is done by ensuring that $\mu(\overline{B}(s,r) \setminus B(s,r)) = 0$ [21]. These new balls are called μ-continuity balls.

A computable Riesz representation theorem is proved in [84, 71]: a measure can be equivalently represented as the positive linear functional mapping continuous real-valued functions with compact support to their integrals. The result is proved for locally compact Hausdorff spaces satisfying computability assumptions, and Borel regular measures.

A measure can alternatively be represented as a valuation on the lower semicontinuous functions [33, 107].

7.2.2.2 Effectively Approximable Sets and Functions

The notion of a computable function expresses the intuitive idea of an algorithm processing an input and producing an output. Computable functions being necessarily continuous, this notion is sometimes too restrictive and one needs a weaker notion, based on the idea that the algorithm performing the computation is allowed to make mistakes on a small set of inputs, in a controlled way. This is the motivation underlying the notion of an effectively approximable function. This definition was introduced by Ko [79] on Euclidean spaces and generalized to other topological spaces by Bosserhoff [21].

Definition 7.2.2. Let X, Y be computable metric spaces and μ a computable Borel probability measure over X. A function $f : X \to Y$ is *effectively μ-approximable* if there exists an oracle Turing machine that given a name of $x \in X$ and a rational $\varepsilon > 0$ outputs a name of $f(x)$ for all x in a set of measure at least $1 - \varepsilon$.

One may require a weaker condition: given ε, δ, the machine computes $f(x)$ at precision δ for all x in a set of measure at least $1 - \varepsilon$. This apparently weaker requirement is actually equivalent to the one given in the definition and is sometimes simpler to prove. Moreover it is sufficient to check this condition only when $\delta = \varepsilon$.

Example 7.2.3 (Random harmonic series). If a binary sequence $s \in \{0,1\}^{\mathbb{N}}$ is obtained by independently tossing a fair coin then the sum

$$f(s) = \sum_{n \geq 1} \frac{(-1)^{s_n}}{n}$$

converges almost surely. Can the sum be computed from s? The function $s \mapsto f(s)$ is not computable because it is obviously not continuous: knowing the n first values of s gives no information about the limit, which can be any real number.

However one can easily prove that the function f is effectively approximable w.r.t. the uniform measure over $\{0,1\}^{\mathbb{N}}$. Let δ, ε be positive rational numbers. For each m, consider the random variable $T_m = \sum_{n>m} (-1)^{s_n}/n$, whose expected value is $\mathbb{E}[T_m] = 0$. Its variance $\mathbb{E}[T_m^2] = \sum_{n>m} 1/n^2$ converges effectively to 0

as m grows, so by Chebyshev's inequality $\mathbb{P}[|T_m| \geq \delta] \leq \mathbb{E}[T_m^2]/\delta^2$ can be taken as small as we want, in particular smaller than ε, by taking m sufficiently large. The sum $\sum_{n=1}^{m}(-1)^{s_n}/n$ is then δ-close to $f(s)$ for all s in a set of measure at least $1-\varepsilon$, and that finite sum can be uniformly computed from s and ε, δ.

Example 7.2.4 (Pólya urn). In an urn starting with one black ball and one white ball, at each round one draws a ball at random and puts it back into the urn together with a new ball of the same color. The sequence of observed colors follows a probability distribution μ over $\{0,1\}^{\mathbb{N}}$ (interpreting 1 as black and 0 as white for instance) defined by

$$\mu([u]) = \frac{(|u|_0)! \cdot (|u|_1)!}{(1+|u|)!},$$

where $u \in \{0,1\}^*$, $|u|_0$ is the number of occurrences of 0 in u and $|u|_1$ is the number of occurrences of 1 in u (0! evaluates to 1 here). For μ-almost every outcome sequence $s \in \{0,1\}^{\mathbb{N}}$, the frequency of 1's in the sequence converges to a real number $p(s)$. Can $p(s)$ be computed from s? Again the function p is heavily discontinuous hence not computable, but one can prove that it is effectively μ-approximable. This can be done as in the previous example by showing that the speed of convergence is effective (this will be formalized in Section 7.2.2.5), or by a more abstract argument that will be presented in Examples 7.2.27 and 7.4.14.

The notion of effective approximability for functions has an immediate counterpart for sets [79].

Definition 7.2.5. Let X be a computable metric space and μ a computable Borel probability measure over X. A set $A \subseteq X$ is *effectively μ-approximable* if its characteristic function $1_A : X \to \{0,1\}$ is effectively μ-approximable.

If $A \subseteq X$ is effectively μ-approximable then $\mu(A)$ is computable.

Example 7.2.6 (Smith-Volterra-Cantor set). The Smith-Volterra-Cantor set, which is also known as the fat Cantor set, is a nowhere dense closed subset of $[0,1]$ that has positive Lebesgue measure. It is obtained by starting from $[0,1]$ and by removing, at each stage $n \geq 1$, subintervals of width $1/4^n$ from the middle of each remaining interval. The limit set has Lebesgue measure $1/2$. This set is effectively λ-approximable, by the next observation.

Proposition 7.2.7 (Ko [79]). *Let A be an effectively closed set or an effectively open set. A is effectively μ-approximable if and only if $\mu(A)$ is computable.*

If $f : X \to [0,+\infty]$ is effectively μ-approximable and *bounded* then $\int f \, \mathrm{d}\mu$ is computable.

If $f : X \to [0,+\infty]$ is effectively μ-approximable but is *unbounded* then $\int f \, \mathrm{d}\mu$ is not necessarily computable, but is always lower semicomputable.

Example 7.2.8 (Non-computable integral). Let $(n_i)_{i\in\mathbb{N}}$ be a computable sequence enumerating a non-computable set $A \subseteq \mathbb{N}$ such as the halting set. The piecewise

constant function $f : [0,1] \to [0,+\infty)$ defined by $f(x) = 2^{-n_i+i+1}$ on $(2^{-i-1}, 2^{-i}]$ and $f(0) = 0$ is effectively λ-approximable (there is an algorithm that computes f outside the null set $\{0\} \cup \{2^{-i-1} : i \in \mathbb{N}\}$) but its integral is $\sum_{n \in A} 2^{-n}$, which is not a computable real number.

Composing effectively approximable functions can be done for an appropriate choice of the involved measures. A measurable function $f : X \to Y$ pushes any measure μ over X to a measure over Y, denoted by μ_f and called the push-forward of μ under f. It is defined by $\mu_f(A) = \mu(f^{-1}(A))$ for all measurable sets A.

Proposition 7.2.9 (Bosserhoff [21]). *If $f : X \to Y$ is effectively μ-approximable then the push-forward measure μ_f is computable. If $g : Y \to Z$ is effectively μ_f-approximable then $g \circ f$ is effectively μ-approximable.*

Proof. To prove that μ_f is computable, we show that if $h : Y \to [0,1]$ is a bounded computable function then $\int h \, \mathrm{d}\mu_f$ is computable, uniformly in h. By definition of μ_f, one has $\int h \, \mathrm{d}\mu_f = \int h \circ f \, \mathrm{d}\mu$. The function $h \circ f$ is effectively μ-approximable (simply compose the algorithms for f and h) so $\int h \circ f \, \mathrm{d}\mu$ is computable as h is bounded. Everything is uniform in h.

If g is effectively μ_f-approximable then the algorithms approximating f and g can easily be composed to approximate $g \circ f$. □

The computational complexity of integration has been investigated by Ko [79] and Kawamura [73] for continuous functions on $[0,1]$ with the Lebesgue measure. They essentially proved that the complexity of integration corresponds to the counting class #P and is in some sense complete for this class. For other measures than the Lebesgue measure, the dependence of the complexity of integration on the measure has been investigated by Férée and Ziegler [35].

7.2.2.3 Effective Measurability

Computable functions can be seen as the effective version of continuous functions: a function f is computable if and only if it is effectively continuous in the sense that the pre-image of effectively open sets are effectively open sets, uniformly; also, a function is continuous if and only if it is computable relative to some oracle.

In the same way, the weaker notion of effectively approximable function can be interpreted as the effective counterpart of a classical notion, measurability. To this end, we briefly present a notion of computability and a representation for measurable sets.

Let X be a computable metric space and μ a computable Borel probability measure over X. The set of Borel subsets of X can be endowed with a pseudometric $d_\mu(A,B) = \mu(A \Delta B)$. The quotient of this pseudometric space by the equivalence relation $A \equiv_\mu B \iff d_\mu(A,B) = 0$ is a separable metric space. We now show how to choose a countable dense subset in order to make it a computable metric space.

The space X has a topological basis of sets whose measures are computable real numbers. This basis is obtained by computing a sequence of positive real numbers $(r_i)_{i \in \mathbb{N}}$ that is dense in $(0,\infty)$, such that $\mu(\overline{B}(s,r_i) \setminus B(s,r_i)) = 0$ for all i and s

in the countable dense subset associated with X (see [21]). The ring $\mathscr{R}$ obtained by taking the closure of this basis under finite unions and complements has a canonical numbering $\mathscr{R} = \{R_i : i \in \mathbb{N}\}$. One can then take $\mathscr{R}$ as a dense sequence in the quotient space of measurable sets.

The computable metric space of (equivalence classes of) measurable sets induces a representation and a notion of computability for measurable sets: A is represented by any Cauchy sequence of ring elements converging fast to A in the pseudometric d_μ. An equivalent representation consists in describing the sequence of real numbers $\mu(A \cap R_i)$.

The next definition and result appeared in [79] on $\mathbb{R}$ and in [21] on general spaces.

Definition 7.2.10. A set A is *effectively μ-measurable* if the following equivalent conditions hold:

- Given a positive rational ε one can compute, uniformly in ε, a finite union A_ε of basic μ-continuity balls such that $\mu(A \Delta A_\varepsilon) \leq \varepsilon$,
- Given i one can compute $\mu(A \cap R_i)$ uniformly in i.

Proposition 7.2.11. *A set is effectively μ-approximable if and only if it is effectively μ-measurable.*

For a proof, one can consult [79, 21]. Other ways of representing measurable sets have been investigated on general measurable spaces with σ-finite measures [120, 119]. One can also define the notion of an effectively μ-measurable function and prove that it is equivalent to effective μ-approximability. We do not include it here as it would require some extra definitions and will not be used in this chapter. The interested reader can consult [79, 118].

7.2.2.4 L^p-Spaces and Absolute Continuity

For each computable real number p, the space $L^p(X,\mu)$ is a complete separable metric space that is naturally a computable metric space. The finite linear combinations with rational coefficients of characteristic functions of elements of $\mathscr{R}$ are dense and their canonical numbering makes the metric

$$d(f,g) = \|f-g\|_p = \left(\int_X |f-g|^p \, \mathrm{d}\mu\right)^{1/p}$$

computable. This structure induces a representation of the elements in $L^p(X,\mu)$, which are equivalence classes of functions under μ-almost everywhere coincidence, and a notion of $L^p(X,\mu)$-computable function.

Definition 7.2.12. A measurable function $f : X \to \mathbb{R}$ is $L^p(X,\mu)$-*computable* if its equivalence class is a computable element of $L^p(X,\mu)$.

Several characterizations of L^p-computable functions have been obtained [99, 122]. L^p-computability is not far away from effective approximability, as the next result shows (an indirect proof appears in [66]).

Proposition 7.2.13. *A function $f : X \to \mathbb{R}$ is $L^p(X,\mu)$-computable if and only if it is effectively μ-approximable and $\|f\|_p$ is computable.*

Any non-negative function in $L^1(X,\mu)$ induces a finite measure ν with density f w.r.t. μ:

$$\nu(A) = \int_A f \,\mathrm{d}\mu$$

(here we assume that $\|f\|_1 = 1$ so that ν is a probability measure). The measure ν is absolutely continuous w.r.t. μ, written $\nu \ll \mu$, which means that for every measurable set A, $\mu(A) = 0$ implies $\nu(A) = 0$. The Radon-Nikodym theorem asserts that every absolutely continuous measure can be obtained this way. The function f is called the Radon-Nikodym derivative of ν w.r.t. μ and is denoted by $\frac{\mathrm{d}\nu}{\mathrm{d}\mu}$.

The computability of the correspondence between absolutely continuous measures and densities in L^1 has been investigated. We present a few results in this direction.

Definition 7.2.14 (Effective absolute continuity). Let μ, ν be Borel probability measures. We say that ν is *effectively absolutely continuous w.r.t.* μ if there exists a computable function $\varphi : \mathbb{N} \to \mathbb{N}$ such that $\mu(A) \leq 2^{-\varphi(n)}$ implies $\nu(A) \leq 2^{-n}$ for all Borel sets A.

In that case, the function mapping μ-measurable sets to ν-measurable sets is well defined and computable: any μ-approximation of A at precision $2^{-\varphi(n)}$ is a ν-approximation of A at precision 2^{-n}. It implies that every effectively μ-approximable set $A \subseteq X$ or function $f : X \to Y$ is also effectively ν-approximable.

Observe that if $\nu \leq c\mu$ for some constant c then ν is effectively absolutely continuous w.r.t. μ and $\frac{\mathrm{d}\nu}{\mathrm{d}\mu}$ is essentially bounded by c. The next result is indirectly proved in [87].

Proposition 7.2.15. *Let μ be a computable probability measure and f be a non-negative $L^1(X,\mu)$-computable function. The measure ν defined by $\frac{\mathrm{d}\nu}{\mathrm{d}\mu} = f$ is computable and effectively absolutely continuous w.r.t. μ.*

Proof. In order to prove that ν is computable, we show that the integral $\int h \,\mathrm{d}\nu = \int fh \,\mathrm{d}\mu$ of any computable bounded function $h : X \to [0,1]$ is uniformly computable. One has $\int f \,\mathrm{d}\mu = \int fh \,\mathrm{d}\mu + \int f(1-h) \,\mathrm{d}\mu$. Both fh and $f(1-h)$ are effectively μ-approximable, so their integrals are lower semicomputable. As their sum is computable, they are computable as well.

We now show that ν is effectively absolutely continuous w.r.t. μ. Given ε, as f is L^1-computable one can effectively find a bounded computable function $h : X \to \mathbb{R}$ such that $\int |f-h| \,\mathrm{d}\mu < \varepsilon/2$. Let N be an upper bound on $|h|$ and $\delta = \varepsilon/(2N)$. If $\mu(A) \leq \delta$ then

$$\nu(A) = \int_A f \,\mathrm{d}\mu = \int_A (f-h) \,\mathrm{d}\mu + \int_A h \,\mathrm{d}\mu \leq \varepsilon/2 + N\mu(A) \leq \varepsilon. \qquad \square$$

However, the Radon-Nikodym theorem is not computable, even assuming effective absolute continuity.

Theorem 7.2.16 (Hoyrup, Rojas, Weihrauch [70]). *There exists a computable probability measure over $[0,1]$, effectively absolutely continuous w.r.t. the Lebesgue measure λ, whose Radon-Nikodym derivative is not $L^1([0,1],\lambda)$-computable.*

As often, the proof uses the technique appearing in the proof of Pour-El and Richard's First Main Theorem [100] stating that a certain class of discontinuous operators do not preserve computability: the discontinuity can be used to encode the halting problem.

Proof. The operator mapping $f \in L^1([0,1],\lambda)$ to the measure ν with density f is continuous (and even computable), but its inverse is not. For instance, the sequence $f_n(x) = 1 + \sin(2\pi nx)$ does not converge to $f_\infty(x) = 1$ in L^1 but the corresponding measures ν_n converge to ν_∞.

Now for $n \in \mathbb{N}$, let $t(n) \in \mathbb{N} \cup \{\infty\}$ be the halting time of Turing machine number n. The sought density function is $f = \sum_{n\geq 1} 2^{-n} f_{t(n)}$. One can easily see that the corresponding measure $\nu = \sum_{n\geq 1} 2^{-n} \nu_{t(n)}$ is computable because the mapping $n \mapsto \nu_{t(n)}$ is computable. However, f is not L^1-computable as $\int |f-1| \, \mathrm{d}\lambda = \frac{2}{\pi} \sum_{n:M_n \text{ halts}} 2^{-n}$ is not a computable real number.

Observe that $|f| \leq 2$, so $\nu \leq 2\lambda$ hence ν is effectively absolutely continuous w.r.t. λ. □

Moreover the operator mapping an absolutely continuous measure to its derivative is strongly Weihrauch equivalent to the operator lim; see the last chapter of this handbook.

7.2.2.5 Effective Convergence

Many theorems in measure and probability theory are about convergence of functions or random variables. There are many types of convergence, the most classical ones being:

- Convergence in L^p-norm,
- Almost sure convergence,
- Convergence in probability,
- Convergence in distribution.

In order to carry out a computable analysis of convergence theorems, one has to define effective versions of these notions. Usually a sequence converges to a limit if for every prescribed precision one can find a rank from which the sequence is close to the limit within that precision. This formulation has an immediate effective version, where the rank can be uniformly computed from the given precision. This is the approach we present here. We will see in Section 7.3 another way of analyzing the computable content of almost sure convergence theorems, using algorithmic randomness. The next definition appeared in [112].

Definition 7.2.17. Let μ be a probability measure over X and $f_n, f : X \to \mathbb{R}$ be measurable functions. We say that f_n converges *effectively μ-almost surely* to f if

$$\forall \varepsilon > 0, \exists n, \quad \mu(\{x \in X : \exists p \geq n, |f_p(x) - f(x)| > \varepsilon\}) \leq \varepsilon \tag{7.1}$$

and $n \in \mathbb{N}$ can be uniformly computed from $\varepsilon \in \mathbb{Q}$.

We say that f_n converges *effectively in probability* to f if

$$\forall \varepsilon > 0, \exists n, \forall p \geq n, \quad \mu(\{x \in X : |f_p(x) - f(x)| > \varepsilon\}) \leq \varepsilon \tag{7.2}$$

and $n \in \mathbb{N}$ can be uniformly computed from $\varepsilon \in \mathbb{Q}$.

Observe that (7.1) is equivalent to the more usual formulation of μ-almost sure convergence

$$\mu(\{x \in X : f_n(x) \text{ converge to } f(x)\}) = 1. \tag{7.3}$$

As in the classical setting, these effective convergence notions are interrelated.

- Effective almost sure convergence implies effective convergence in probability,
- Effective convergence in L^p-norm implies effective convergence in probability,
- When the sequence is bounded by an L^p-computable function, effective convergence in probability implies (therefore, is equivalent to) effective L^p-convergence.

We will see that many convergence theorems from probability theory are not computable, in the sense that the convergence is not effective in any sense (Theorems 7.2.21, 7.3.7, 7.3.8). Such negative results can often be proved by showing that the limit is not effectively approximable, thanks to the following result, appearing in [67].

Proposition 7.2.18. *Let $f_n : X \to \mathbb{R}$ be uniformly effectively μ-approximable. If they converge effectively in probability to $f : X \to \mathbb{R}$ then f is effectively μ-approximable.*

Proof. To compute f with probability of error δ and at precision ε, compute f_n at precision $\varepsilon/2$ with probability of error δ, where n is associated with $\varepsilon/2$ in the effective convergence. □

The proof of this proposition is essentially the argument that we used to show that the limit of the random harmonic series is effectively approximable (Example 7.2.3).

7.2.3 Results in Computable Measure and Probability Theory

Many results from measure theory have been investigated in computable analysis. Some of them are about the computability of certain measures, others are about the computability of convergence theorems.

The operation of conditioning a measure is a fundamental construct in measure theory. Its (non-)computability has been investigated in [1, 2].

In addition to the results presented here, the computability of invariant measures of dynamical systems has been investigated in [51, 56]. The problem of computing pseudo-random points, i.e. points satisfying prescribed properties that hold almost surely, has been investigated in [13, 52, 54].

7.2.3.1 Conditioning

Conditioning is a fundamental concept in probability theory. Its (non-)computability has been investigated by Ackerman, Freer and Roy [1, 2]. They prove that the operation of conditioning a probability measure is not computable in general. We present a similar result, whose proof is based on the non-computability of the Radon-Nikodym theorem (Theorem 7.2.16).

Let π is a Borel probability measure over a product space $X \times Y$ and π_X its marginal measure over X defined by $\pi_X(A) = \pi(A \times Y)$ for Borel sets $A \subseteq X$. One can define the conditional probability measures $\pi(.|x)$ over Y for π_X-almost all $x \in X$, such that π is a combination of these measures: $\pi(A \times B) = \int_A \pi(B|x)\, \mathrm{d}\pi_X(x)$. The function from X to $\mathscr{M}_1(X)$ mapping x to $\pi(.|x)$ is called a *disintegration* of π. It is not unique, but two such disintegrations must agree π_X-almost everywhere. Usually these mappings are not continuous, so they cannot be computable. In [1] it is proved that even when a disintegration is discontinuous on a set of measure 1, it need not be computable on a set of measure 1. In [2] it is proved that even when there is a unique continuous disintegration, it need not be computable. The non-computability is also expressed in terms of Weihrauch degrees: the disintegration operator is strongly Weihrauch equivalent to lim; see the last chapter of this handbook.

We present another example, based on the non-computability of the Radon-Nikodym derivative. Indeed, conditional probabilities are usually constructed using the Radon-Nikodym theorem, thus Theorem 7.2.16 immediately implies the non-computability of conditional probabilities.

Theorem 7.2.19. *Let $X = [0,1]$ and $Y = \{0,1\}$. There is a computable measure π over $X \times Y$ whose disintegration $x \mapsto \pi(0|x)$ is not effectively π_X-approximable, where π_X is the marginal measure over X (and is the Lebesgue measure λ here).*

Proof. Let ν be the computable probability measure over $[0,1]$ from the proof of Theorem 7.2.16: $\nu \ll \lambda$ and even $\nu \leq 2\lambda$, but $\frac{\mathrm{d}\nu}{\mathrm{d}\lambda}$ is not $L^1(\lambda)$-computable. As $\frac{\mathrm{d}\nu}{\mathrm{d}\lambda}$ is bounded by 2, $\frac{\mathrm{d}\nu}{\mathrm{d}\lambda}$ is not effectively λ-approximable.

As $\nu \leq 2\lambda$, one has $\lambda = \frac{1}{2}(\nu + \mu)$ where μ is another computable probability measure (μ is simply defined as $2\lambda - \nu$). Consider the measure π over $X \times Y$ defined by $\pi(A \times \{0\}) = \frac{1}{2}\nu(A)$ and $\pi(A \times \{1\}) = \frac{1}{2}\mu(A)$. π is a computable measure; its marginal measures are the uniform measures over X and Y. The conditional expectation is $\pi(0|x) = \frac{\mathrm{d}\nu}{\mathrm{d}\lambda}(x)$ for λ-almost every $x \in [0,1]$. □

7.2.3.2 Birkhoff Ergodic Theorem

One of the most celebrated results in probability theory is the Birkhoff ergodic theorem generalizing the strong law of large numbers from independent random variables to stationary ones.

Ergodic theory is a branch of dynamical systems that focuses on the global properties of dynamical systems (we refer to the introductory book [25]). A (discrete-time) dynamical system is just a set X and a function $T : X \to X$ acting on X.

Points in X are the possible states of the system, and $T(x)$ is the state of the system at time $t+1$ if x is the state at time t. The orbit of a point x is the sequence $x, T(x), T^2(x), \ldots$, which is simply the evolution of the system over time when starting in state x. Ergodic theory enables one to describe how the orbits of the system are distributed over X.

To do this, one needs some structure on the set X. Assume that X is a measurable space and μ is a probability measure over X. A measurable transformation $T : X \to X$ *preserves* μ if for every Borel set A, $\mu(T^{-1}(A)) = \mu(A)$. We can alternatively say that μ is T*-invariant*. Intuitively, applying T to points of X does not change their distribution over the space. The Birkhoff ergodic theorem states that if T preserves μ then for μ-almost every x, the asymptotic distribution of the orbit of x under T converges.

Theorem 7.2.20 (Birkhoff ergodic theorem). *If $T : X \to X$ preserves μ then for every $f \in L^1(X, \mu)$, the limit*

$$f^*(x) := \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} f \circ T^i(x) \tag{7.4}$$

exists for μ-almost all $x \in X$. Moreover $f^* \in L^1(\mu)$ and $\int f^* \, \mathrm{d}\mu = \int f \, \mathrm{d}\mu$.

The averages in (7.4) are called the Birkhoff averages of f.

The case when f is the characteristic function of a measurable set $A \subseteq X$ is particularly suggestive: for μ-almost every x, the visiting frequency of its orbit in A converges, so it is indeed about the distribution of its orbit in the space.

This is an almost sure convergence theorem. Computing or quantifying the speed of convergence of this theorem has been a longstanding problem, already investigated by ergodic theorists. Kakutani and Petersen [72] proved that there is no general bound on the speed of convergence in the ergodic theorem, but their result does not formally exclude the possibility of a computable speed. They construct very irregular functions f making the convergence as slow as wanted, but it may happen that for simple functions f, the speed of convergence can be estimated. Bishop [19] informally argued that the ergodic theorem is nonconstructive, thus suggesting that it is not computable in any sense. This was made precise by V'yugin [112], who proved that the speed of convergence is indeed not computable in general.

V'yugin's example is based on one of the simplest dynamical systems, the shift operator from the Cantor space $X = \{0,1\}^{\mathbb{N}}$ to itself mapping a sequence $x_0x_1x_2\ldots$ to $x_1x_2x_3\ldots$, the same sequence with the first bit removed. The function $f : X \to [0,1]$ is very simply defined by $f(x_0x_1x_2\ldots) = x_0$.

Theorem 7.2.21 (V'yugin [112]). *On the Cantor space $X = \{0,1\}^{\mathbb{N}}$, there is a computable shift-invariant probability measure μ such that the convergence of the Birkhoff averages of $f : x \mapsto x_0$ is not effective in any sense (almost sure, in probability, in L^1-norm).*

Proof. We give the proof because it motivates the next discussions. We present a slight variant of V'yugin's original proof. It again follows the argument of Pour-El

and Richard's First Main Theorem, already met in Theorem 7.2.16. Informally, the operator mapping μ to $f^* \in L^1(X,\mu)$ is "not continuous".

Given $p \in [0,1]$, consider the measure μ_p over $\{0,1\}^{\mathbb{N}}$ defined as the distribution of the following random infinite binary sequence. Take $x_0 \in \{0,1\}$ uniformly at random. Once x_i has been drawn, let $x_{i+1} = 1 - x_i$ with probability p, and $x_{i+1} = x_i$ with probability $1-p$.

If $p = 0$ then the sequence will contain only 0's with probability $1/2$, and only 1's with probability $1/2$. If $p > 0$ then the sequence will almost surely contain infinitely many blocks of 0's and 1's, very long if p is close to 0.

The discontinuity comes from the fact that f^* has μ_p-almost surely value $\frac{1}{2}$ for $p > 0$ but has μ_p-almost surely values 0 and 1 for $p = 0$.

We can now encode the halting problem using this discontinuity. For $n \in \mathbb{N}$, let $t(n) \in \mathbb{N} \cup \{\infty\}$ be the halting time of Turing machine number n. The sought measure is

$$\mu = \sum_{n \geq 1} 2^{-n} \mu_{2^{-t(n)}}. \tag{7.5}$$

One can easily see that μ is computable because the mapping $n \mapsto \mu_{2^{-t(n)}}$ is computable. However, f^* is not $L^1(X,\mu)$-computable, which implies that the convergence of the Birkhoff averages to f^* is not computable in any sense (see Proposition 7.2.18). Proving that f^* is not a computable element of $L^1(\mu)$ is easy: otherwise it would be effectively μ-approximable by Proposition 7.2.13, from which one derives that the singleton $\{0000\ldots\}$ is effectively μ-approximable, which is not possible as $\mu(\{0000\ldots\}) = \sum_{n:M_n \text{ does not halt}} 2^{-n}$ is not a computable real number. □

In this argument one sees that the asymptotic distribution of the orbit of x depends very much on x: the orbits of $\overline{0}$ and $\overline{1}$ are each concentrated in one point while the orbits of other points are densely distributed in the space. The non-computability comes from here, as one cannot computably distinguish between these different behaviors.

The fact that several different distributions are possible is expressed as the system being *non-ergodic*. More precisely, one says that the system (X,μ,T) is *ergodic* if the only measurable sets A such that $T^{-1}(A) = A$ are trivial, i.e. have measure 0 or 1. Note that such sets are stable under T, so orbits starting inside A or outside A may have very different behaviors. When the system is ergodic, the limit function f^* is constant μ-almost everywhere, which corresponds to the intuition that almost all the orbits have the same distribution. If (X,T) is fixed, we say that μ is ergodic if the system (X,T,μ) is ergodic.

Let us go back to V'yugin's result. The constructed measure μ is *not* ergodic, notably because the limit function f^* is not constant μ-almost everywhere. This is actually mandatory to make the convergence non-computable, as the next result shows. Here X is a computable metric space, μ is a computable Borel probability measure over X and $T : X \to X$ is computable and μ-invariant.

Theorem 7.2.22 (Avigad, Gerhardy, Towsner [8]). *If the system (X,μ,T) is computable and ergodic and f is a computable element of $L^1(\mu)$ then the speed of convergence of the Birkhoff averages of f is computable.*

They actually prove that in the general (i.e. non-ergodic) case, the speed of convergence is always computable relative to the $L^2(\mu)$-norm of f^*, assuming $f \in L^2(\mu)$. A simpler proof in the ergodic case can be found in [53].

7.2.3.3 Ergodic Decomposition

We saw that the ergodic systems are those systems for which almost all orbits have the same distribution. If a system is not ergodic then two orbits may have very different asymptotic properties. But if one groups together all the points whose orbits have a given distribution then these points form an ergodic subsystem. Indeed a non-ergodic system can always be decomposed into disjoint ergodic subsystems. This is expressed by the following result, which is an application of the Choquet theorem from convex analysis [98].

Theorem 7.2.23 (Ergodic decomposition theorem). *If μ is a T-invariant probability measure then there exists a unique probability measure m over the class of measures such that:*

- *m gives measure 1 to the set of ergodic T-invariant measures*
- *μ is the barycenter of m, i.e. $\mu(A) = \int \nu(A)\,\mathrm{d}m(\nu)$ for all measurable sets A.*

The ergodic measures can be equivalently defined as the invariant measures that cannot be expressed as combinations of invariant measures, except the trivial one where m is the Dirac measure concentrated on μ.

For instance in V'yugin's construction, the measure μ is a combination of countably many ergodic measures: the Dirac measures $\delta_{000\ldots}$ and $\delta_{111\ldots}$ and for each $t \in \mathbb{N}$, the measure with parameter $p = 2^{-t}$. The decomposition (7.5) $\sum_{n\geq 1} 2^{-n}\mu_{2^{-t(n)}}$ is computable, but *is not* the ergodic decomposition of μ: for $t(n) = \infty$, one has $\mu_{2^{-t(n)}} = \mu_0 = \frac{1}{2}\delta_{000\ldots} + \frac{1}{2}\delta_{111\ldots}$, which is not ergodic as it can be further decomposed as a combination of two ergodic measures $\delta_{000\ldots}$ and $\delta_{111\ldots}$. The corresponding ergodic decomposition of μ turns out to be non-computable, and this is necessary and sufficient for the counterexample to work, as the following result shows.

Proposition 7.2.24 (Hoyrup [64]). *Let T be computable and μ be a computable T-invariant measure. The following statements are equivalent:*

- *The μ-almost sure convergence of the Birkhoff averages of bounded computable functions is effective,*
- *The ergodic decomposition m of μ is computable.*

Observe that an invariant measure is usually decomposed into continuously many ergodic measures. In V'yugin's construction, the ergodic decomposition of μ is countably infinite. Is it possible to build another example with a finite number of ergodic measures only? The answer is positive, but one needs a very different argument.

Theorem 7.2.25 (Hoyrup [65]). *There exists a computable shift-invariant measure μ whose ergodic decomposition is $\mu = \frac{1}{2}\mu_0 + \frac{1}{2}\mu_1$, where μ_0 and μ_1 are non-computable ergodic measures.*

The proof borrows a construction scheme from computability theory, namely the priority method with finite injury.

There are interesting classes of invariant measures for which the ergodic decomposition is computable [62].

Theorem 7.2.26. *Let $\mathscr{C} \subseteq \mathscr{M}_1(X)$ be an effectively compact class of ergodic measures. If μ is a computable measure which is a convex combination of measures in $\mathscr{C}$ then the ergodic decomposition of μ is computable.*

Proof. By the assumption on $\mathscr{C}$, the class of Borel probability measures supported in $\mathscr{M}_1(\mathscr{C})$ is effectively compact in the computable metric space $\mathscr{M}_1(\mathscr{M}_1(X))$ of measures over $\mathscr{M}_1(X)$. The combination operator mapping $m \in \mathscr{M}_1(\mathscr{M}_1(C))$ to $\mu \in \mathscr{M}_1(X)$ is computable and one-to-one, so its inverse is computable. □

Example 7.2.27 (Pólya urn continued). The class of Bernoulli measures over $\{0,1\}^{\mathbb{N}}$ is an example of an effectively compact class of ergodic measures. In the Pólya urn model from Example 7.2.4, the measure μ is a combination of Bernoulli measures and is computable, so by Theorem 7.2.26 its decomposition is computable. As a Bernoulli measure can be (computably) identified with its parameter in $[0,1]$, it means that the distribution of the function $p(s)$ from Example 7.2.4 is a computable measure over $[0,1]$.

A Bernoulli measure over $\{0,1\}^{\mathbb{N}}$ is the joint distribution of a sequence of i.i.d. random variables in $\{0,1\}$. They can be generalized by considering the distribution of a sequence of i.i.d. random variables in $\mathbb{R}$. These distributions are exactly the product measures $\nu^\infty := \bigotimes_{i\in\mathbb{N}} \nu$, where ν is any probability measure over $\mathbb{R}$. This class of measures $\mathscr{C} = \{\nu^\infty : \nu \in \mathscr{M}_1(\mathbb{R})\} \subseteq \mathscr{M}_1(\mathbb{R}^{\mathbb{N}})$ is no longer effectively compact. However Freer and Roy proved a version of Theorem 7.2.26 for this class of measures.

Theorem 7.2.28 (Freer, Roy [47]). *If μ is a computable measure which is a combination of measures in $\mathscr{C}$ then the decomposition of μ is computable.*

De Finetti's theorem states that the convex combinations of measures in $\mathscr{C}$ are exactly the joint distributions of exchangeable sequences of random variables in $\mathbb{R}$, so Theorem 7.2.28 can be reformulated as follows: if the joint distribution μ of an exchangeable sequence of random variables in $\mathbb{R}$ is computable, then the unique measure m over $\mathscr{C}$ such that $\mu(A) = \int \nu(A)\,\mathrm{d}m(\nu)$ is computable.

7.3 Algorithmic Randomness

Probability theory provides many important convergence theorems, notably almost sure convergence theorems such as the strong law of large numbers, Birkhoff's er-

godic theorem, the martingale convergence theorems and the Lebesgue differentiation theorem.

In this section, we will work in a probability space, but many of the ideas extend to other types of measures. Recall that a sequence $(f_n)_{n\in\mathbb{N}}$ of random variables converges almost surely to f if $f_n(x)$ converges to $f(x)$ for almost every x. How does one approach the computability of such a result? The option adopted in Definition 7.2.17 is to use the following equivalent formulation:

$$\forall \varepsilon > 0, \exists n, \quad \mathbb{P}[\forall p \geq n, |f_p(x) - f(x)| \leq \varepsilon] \geq 1 - \varepsilon.$$

This is an existence statement, stating the existence of n given ε, and as such it can be analyzed from a computability perspective. We already saw computability results about almost sure convergence theorems, which happened to be negative for the most part: one cannot in general compute the speed of convergence.

There is another way of formulating almost sure convergence as an existence result:

$$\exists A, \mathbb{P}[A] = 1 \text{ and } \forall x \in A, f_n(x) \text{ converges to } f(x). \tag{7.6}$$

How does one investigate the computability of such a result? What does it mean to compute the set A? As A has measure 1, it is trivially an effectively measurable set in the sense of Section 7.2.2.3. Indeed, being effectively measurable is not about a set of points, but about its equivalence class. As a result, one needs a finer effective notion of measurable sets, in particular a notion of effective sets of full measure and effective null sets. This is one of the successes of algorithmic randomness. The reference books in this field are [82, 95, 32], where the theory is developed on the Cantor space. For the extension to more general spaces, we refer the reader to [58, 59, 49, 68, 88, 103, 104].

7.3.1 Effective Null Sets

A null set is a set of measure zero. As we are working with regular measures, another way to characterize a null set is to say that a set N is null if and only if for every $\varepsilon > 0$, the set N can be covered by an open set of measure less than ε. Martin-Löf [85] noticed that this characterization of null set can be effectivized.

We are working in any computable metric space X endowed with a computable Borel probability measure μ over X. For short, we say that (X, μ) is a *computable probability space*. As is usual in probability theory, we sometimes write $\mathbb{P}[A]$ for $\mu(A)$, where $A \subseteq X$ is any Borel set.

Definition 7.3.1. A *Martin-Löf test* is a sequence $(U_n)_{n\in\mathbb{N}}$ of uniformly effectively open sets such that $\mu(U_n) \leq 2^{-n}$. A set N is *Martin-Löf null* if $N \subseteq \bigcap_n U_n$ for some Martin-Löf test. A point x is *Martin-Löf random* if it is not contained in any Martin-Löf null set. The set of Martin-Löf random points is written ML.

As the name suggests, Martin-Löf random points generally have "typical" or "random" behavior. For example, consider a Martin-Löf random sequence in the space of fair-coin tosses. Such a sequence will always satisfy standard randomness criteria such as the strong law of large numbers and the law of the iterated logarithm. Indeed, it is difficult to come up with an almost sure property of random coin flips which is not true of a Martin-Löf random sequence, and such properties are almost always computability-theoretic in nature. (For example, there is a Martin-Löf random sequence which computes the halting problem.) In this definition we have assumed the measure μ to be computable. However it can be extended to non-computable measures by requiring the sets U_n to be effective with oracle μ (we define it below, see Definition 7.3.16). When the measure μ is not clear from the context, we speak about μ-tests, μ-null sets and μ-random points, whose set is denoted by $\mathsf{ML}(\mu)$.

Nonetheless, for good reasons other randomness notions have also been explored. While there are many options, we give the six that are most connected to computable measure theory. A *2-Martin-Löf test* is the same as a Martin-Löf test except that the sequence $(U_n)_{n\in\mathbb{N}}$ is allowed to be computable relative to the halting problem. The corresponding notion of randomness is called 2-*randomness*. A *weak* 2-*test* is a null Π^0_2 set, that is a null set which is the intersection of a sequence of uniformly effectively open sets. The corresponding randomness notion is called *weak* 2-*randomness*. Both of these are stronger notions than Martin-Löf randomness (also known as 1-randomness).

The other three notions are weaker. A *Schnorr test* is a Martin-Löf test $(U_n)_{n\in\mathbb{N}}$ where $\mu(U_n)$ is computable uniformly in n. The corresponding notion of randomness is called *Schnorr randomness*. This definition is modeled after Brouwer's definition of a constructive null set. While weaker than Martin-Löf randomness, Schnorr randomness has a tight relationship with computable measure theory.

Computable randomness is a randomness notion that stands strictly between Schnorr and Martin-Löf randomness. A test[1] for computable randomness is a Martin-Löf test $(U_n)_{n\in\mathbb{N}}$ which is "bounded" by some computable measure ν. Specifically, $\mu(U_n \cap A) \leq \nu(A) \cdot 2^{-n}$ for all Borel measurable sets A.

Finally, *Kurtz randomness* (or *weak randomness*) is the weakest randomness notion of the six. A Σ^0_2 set is a computable union of effectively closed sets. A *Kurtz test* is a null Σ^0_2 set. A set is *effectively Kurtz null* if it is a subset of a Kurtz test. A point is *Kurtz random* if it is not contained in an effectively Kurtz null set. Kurtz randomness is much weaker than Schnorr randomness, and many do not consider it to be a true randomness notion. It does not satisfy the law of large numbers, and

[1] On the Cantor space, tests are usually expressed in terms of computable martingales. On general metric spaces, the equivalent definition given here is easier to express.

it shares as many similarities with effective genericity as it does with randomness.[2] Nonetheless, it is useful to consider Kurtz randomness.

To summarize, we have the following algorithmic randomness notions listed from weakest to strongest: Kurtz, Schnorr, computable, Martin-Löf, weak 2- and 2-randomness. While the majority of the work in algorithmic randomness has focused on the Cantor space with the fair-coin (a.k.a. Lebesgue) measure (or in some cases Bernoulli measures), all of these randomness notions naturally extend to other computable metric spaces endowed with a Borel probability measure. One natural candidate is Brownian motion.

7.3.1.1 Brownian Motion

Informally, Brownian motion is a process which resembles a continuous-time random walk starting at the origin. More formally, one can represent d-dimensional Brownian motion as a particular probability measure, called the Wiener measure, on the space $\mathscr{C}([0,1],\mathbb{R}^d)$ (or $\mathscr{C}([0,\infty),\mathbb{R}^d)$) of continuous functions. This space is a computable metric space under the sup norm. Also the Wiener measure is a computable probability measure on this space. Algorithmically random Brownian motion paths have been thoroughly studied by Fouché and others [5, 36, 37, 77, 38, 39, 28, 3, 40, 42, 41].

To give a sense of the techniques involved let us consider recurrence and transience of Brownian motion in two and three dimensions. (This theorem is sometimes stated as "A drunk person will always find their way home, while a drunk bird never will.")

Theorem 7.3.2. *Let $B\colon [0,\infty) \to \mathbb{R}^d$ be a continuous function.*

1. *If $d = 2$ and B is Kurtz random, then B obeys the following recurrence result: for every ε and every $t_0 \geq 0$, there is some $t > t_0$ such that $|B(t)| < \varepsilon$.*
2. *If $d \geq 3$ and B is Schnorr random, then B obeys the following transience result: $|B(t)| \to \infty$ as $t \to \infty$.*

Proof. In both cases, our goal is to analyze the corresponding null set. Also, to avoid "reinventing the wheel" we will use known results in Brownian motion, including the classical theorems we are attempting to effectivize.

For (1), consider a non-recurrent function B in two dimensions. Then there are natural numbers m and n such that $|B(t)| \geq 2^{-n}$ for all $t \geq m$. By standard techniques

[2] This claim that Kurtz randomness is not really a randomness notion can be made formal by considering the relativized versions of the tests. Every null set is a Martin-Löf null set relative to some oracle. Said another way, the only difference between the definition of a null set and a Martin-Löf null set is computability. The same is true for 2-randomness, weak 2-randomness, computable randomness and Schnorr randomness. However, this is not true for Kurtz randomness. The relativized notion of a Kurtz null set is a null F_σ set. It is well known in measure theory that there are null sets which are not contained in any null F_σ set. Indeed, every null F_σ set is meager, which explains the connections with genericity. There still is a connection with analysis. Null F_σ sets are the type of null set associated with Jordan-Peano measurable sets and Riemann integrable functions, as opposed to Lebesgue measurable sets and Lebesgue integrable functions.

in computable analysis this is a Σ^0_2 property, i.e. its complement is the intersection of a sequence of uniformly effectively open sets. By the recurrence theorem on two-dimensional Brownian motion, the set of such paths is null. Therefore, we have the desired Kurtz null test.

For (2), consider a non-transient function B in three dimensions. If B is Kurtz random, then by the same argument as in the recurrent case, $|B(t)|$ is unbounded. Therefore, for every $0 < r < R$, there is a $t_0 > 0$ such that $|B(t_0)| = R$ and a $t_1 > t_0$ such that $|B(t_1)| = r$. Now we will use the following quantitative estimate from probability theory:

$$\mathbb{P}\underbrace{\{B : \exists t_1 > t_0 > 0\ [|B(t_0)| > R \text{ and } |B(t_1)| < r]\}}_{U_{r,R}} \leq \frac{r^{d-2}}{R^{d-2}}.$$

A quick observation is that $U_{r,R}$ is effectively open uniformly in r and R (which we can assume are rationals). Then $\bigcap_{r,R} U_{r,R}$ is a null Π^0_2 set. Therefore, every weak 2-random Brownian motion path is transient. We can do better. Since we have a computable upper bound on $\mathbb{P}[U_{r,R}]$ by picking a sufficiently fast-shrinking/growing set of pairs $r_n \to 0$ and $R_n \to \infty$, we have a Martin-Löf test $U_n = U_{r_n,R_n}$.

This is the point where most of the results in the literature stop. However, with a little more work we can extend our result to Schnorr randomness by showing that $\mathbb{P}[U_{r,R}]$ is computable in r and R. A common misconception is that we require an *exact formula* here. Indeed, we only require an *algorithm* which converges to the value of $\mathbb{P}[U_{r,R}]$. We provide such an algorithm. Since $U_{r,R}$ is effectively open, $\mathbb{P}[U_{r,R}]$ is lower semicomputable. Therefore it remains to show that $\mathbb{P}[U_{r,R}]$ is upper semicomputable. Choose a large $S > R$ and a small $\varepsilon > 0$. By transience, after $|B(t)| = R$, then almost surely, eventually there is some $t_1 > t$ such that $|B(t_1)| > S$. Let $V_{r,R,S}$ be the set of paths which after hitting R, hit S before r. This set is also effectively open, and therefore $\mathbb{P}[V_{r,R,S}]$ is lower semicomputable.[3] Now, almost every path is in either $U_{r,R}$ or $V_{r,R,S}$ (or both!). However, the overlap of the two sets is small since $U_{r,R} \cap V_{r,R,S} \subset U_{r,S}$. By choosing S large enough we can compute $\mathbb{P}[U_{r,R}]$ to any desired precision. □

The previous proof is typical for this type of result. Most of the results for Martin-Löf randomness in the literature of this type can be strengthened to Schnorr randomness with some additional work.[4]

[3] Formally $V_{r,R,S} = \{B : \exists t_0 < t_1\ [r < |B(t_0)| < R \quad \wedge \quad S < B(t_1) \quad \wedge \quad \min_{t\in[t_0,t_1]} |B(t)| > r]\}$. This is effectively open since we are using $<$ signs and since the the minimum of a continuous function over a closed interval is computable. Also, if we replaced any of the $<$ with $\leq$ it would not change the measure of the set in this case.

[4] By "this type of result", we mean an almost everywhere result for some computable probability space *which does not explicitly refer to computable objects*. Effective convergence results (see the next section) explicitly refer to computable objects and may hold for Martin-Löf randomness but not Schnorr randomness.

7.3.2 Effective Convergence Theorems

As explained at the beginning of Section 7.3, algorithmic randomness provides a way to analyze the computability of almost sure convergence theorems, by identifying the notion of effective null set associated with each such theorem. There has been increasing interest in the past few years in obtaining characterizations of almost sure convergence theorems using notions of algorithmic randomness. We present here some of the most prominent results in the literature, characterizing Martin-Löf, computable and Schnorr randomness using Birkhoff's ergodic theorem and differentiation theorems, but there are other works calibrating the notions of randomness to fit almost sure theorems with varying computability assumptions.

The first results in this direction were obtained very early by characterizing algorithmic randomness notions in terms of convergence of martingales. On the Cantor space, Schnorr proved that the Martin-Löf random points are exactly the points where every *lower semicomputable* martingale is bounded. He introduced computable random points as the points at which every *computable* martingale is bounded. He also introduced what is now called Schnorr randomness, also originally expressed in terms of martingales. We do not give the details of these characterizations, which can be found in the reference books [82], [95] or [32]. Schnorr's work on this topic appeared in his book [106].

7.3.2.1 Ergodic Theorems

As we saw in Section 7.2.3.2, Birkhoff's ergodic theorem is not computable in general, i.e. the convergence is not effective in any sense. What about the effectiveness of the associated null set? In other words, for which class of random points does Birkhoff's ergodic theorem hold?

Bishop already observed that this theorem does not hold constructively. He proposed a constructive proof whose conclusion is classically equivalent to almost sure convergence, but has less constructive content. This statement involves so-called upcrossing inequalities. This idea was later used by V'yugin to prove that the Birkhoff ergodic theorem holds for Martin-Löf random points.

Theorem 7.3.3 (V'yugin [112]). *Let X be a computable metric space, $T : X \to X$ a computable function, μ a T-preserving probability measure over X and $f : X \to \mathbb{R}$ a computable function. For every Martin-Löf μ-random point x, the limit*

$$f^*(x) := \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} f \circ T^i(x)$$

exists.

In other words, the associated null set is a Martin-Löf null set. There has been work investigating the classes of functions f to which Theorem 7.3.3 can be extended. It holds for all bounded continuous (not necessarily computable) func-

tions f, just because computable functions are in a sense dense among them. It was proved in [14, 43, 15] that it also holds when the system is ergodic and $f : X \to [0, +\infty]$ is lower semicomputable.

The result was used in [63] to prove another famous result from ergodic theory, the Shannon-McMillan-Breiman theorem, for Martin-Löf random points. That result was already implicitly proved in [60] using upcrossing inequalities. We will see in Section 7.4.5 that Theorem 7.3.3 can be extended to other classes of functions T, f that are not continuous (layerwise computable functions). It is still open whether that result holds for lower semicomputable $f : X \to [0, +\infty]$ when the system is not ergodic, and for upper semicomputable $f : X \to [0, +\infty]$ (both for ergodic and non-ergodic systems). It is proved in [89] that this result holds for a stronger notion of randomness called Oberwolfach randomness.

A converse to Theorem 7.3.3 was later proved by Franklin and Towsner [44], showing that Martin-Löf randomness is the right notion in that case.

Theorem 7.3.4 (Franklin, Towsner [44]). *If x is not Martin-Löf λ-random then there exists a computable λ-preserving map $T : \{0,1\}^{\mathbb{N}} \to \{0,1\}^{\mathbb{N}}$ and an effectively open set A such that $\lambda(A)$ is computable and such that the Birkhoff averages of $\mathbf{1}_A$ do not converge at x.*

In other words, if a class of effective null sets induces a notion of randomness that is not stronger than Martin-Löf randomness (if for instance it is strictly weaker than Martin-Löf randomness, like Schnorr or computable randomness), then the Birkhoff ergodic theorem is not in general effective for this notion of null set.

We saw that in the ergodic case, the speed of convergence is computable. This has a consequence on the corresponding randomness notion.

Theorem 7.3.5 (Gács, Hoyrup, Rojas [50]). *Let X be a computable metric space and μ a computable probability measure over X. A point x is Schnorr μ-random if and only if for every μ-preserving computable $T : X \to X$ and every bounded continuous $f : X \to \mathbb{R}$ the limit*

$$f^*(x) := \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} f \circ T^i(x)$$

exists.

In other words, if a class of effective null sets induces a notion of randomness that is not stronger than Schnorr randomness then the Birkhoff ergodic theorem for ergodic measures is not in general effective for this notion of null set.

7.3.2.2 Differentiation Theorems

In this section the underlying measure is the Lebesgue or uniform measure over the real interval $[0, 1]$. Let us start with a computable theorem, the Lebesgue differentiation theorem.

Theorem 7.3.6 (Pathak, Rojas, Simpson [97]; Freer, Kjos-Hanssen, Nies, Stephan [45]). *For a real $x \in [0,1]$ the following statements are equivalent:*

- *x is Schnorr random,*
- *For every L^1-computable function $f : [0,1] \to \mathbb{R}$,*

$$\lim_{r \to 0} \frac{1}{\lambda(B(x,r))} \int_{B(x,r)} f \, \mathrm{d}\lambda \text{ exists.}$$

As already mentioned, when considering almost sure convergence theorems one quickly runs into non-computability results. The following theorems are not computable in the sense that the almost sure convergence is not effective. How to distinguish between different types of non-computability? Again, algorithmic randomness allows for a finer look by providing several notions of randomness via several notions of effective null sets.

Theorem 7.3.7 (Brattka, Miller, Nies [24]; Freer, Kjos-Hanssen, Nies, Stephan [45]). *For a real $x \in [0,1]$ the following statements are equivalent:*

- *x is computably random,*
- *Every non-decreasing computable function $f : [0,1] \to \mathbb{R}$ is differentiable at x.*
- *Every computable Lipschitz function $f : [0,1] \to \mathbb{R}$ is differentiable at x.*

Nies [96] proved that this theorem also holds for polynomial randomness and polynomial-time computable functions f.

Theorem 7.3.8 (Demuth [29]; Brattka, Miller, Nies [24]). *For a real $x \in [0,1]$ the following statements are equivalent:*

- *x is Martin-Löf random,*
- *Every computable function $f : [0,1] \to \mathbb{R}$ of bounded variation is differentiable at x.*

This result was obtained by Demuth [29] in the context of constructive analysis and was reformulated in [24] in modern language.

These results highlight the non-computability of some theorems from real analysis. For instance the Jordan decomposition theorem states that every function of bounded variation is a difference of two non-decreasing functions. The two previous theorems witness that this decomposition is not computable in general. These results are also indirect proofs that the convergence in these theorems is not effective, otherwise convergence would occur at every Schnorr random point.

7.3.3 Randomness Preservation

While "almost sure" results are an important category of results in measure theory, there are also many results which relate the behavior of random variables (or

measurable sets) on one space to random variables (or measurable sets) on another space. For example, in probability theory, a random variable representing a sequence of independent fair coin flips can easily be transformed into a random variable representing a random walk on a lattice or a normally distributed random variable. In the context of algorithmic randomness, one wants to be sure that if x is random on one space, then this transformation of x remains random in the corresponding space. From the perspective of computable analysis, we are saying that these transformations preserve effective null sets. Here we give examples of such results.

Let X and Y be computable metric spaces and μ a computable Borel probability measure over X. We recall from Section 7.2.2.2 that a function $f : X \to Y$ induces a measure μ_f over Y called the push-forward measure, defined by $\mu_f(A) = \mu(f^{-1}(A))$ for all Borel sets $A \subseteq Y$. The next results are folklore results.

Theorem 7.3.9 (Randomness preservation). *Let $f : X \to Y$ be a computable function. If x is Martin-Löf μ-random, then $f(x)$ is Martin-Löf μ_f-random.*

This result is well known and holds for Kurtz, Schnorr, Martin-Löf, weak 2- and 2-randomness. However, it does not hold for computable randomness (see Rute [103]). There is also a partial converse to randomness preservation.

Theorem 7.3.10 (No-randomness-from-nothing). *Let $f : X \to Y$ be a computable function. If y is Martin-Löf μ_f-random, then there exists a Martin-Löf μ-random x such that $f(x) = y$.*

No-randomness-from-nothing holds for computable, Martin-Löf, weak 2- and 2-randomness, but not for Schnorr randomness (see Rute [103]). Nonetheless, we will see a result below (Theorem 7.3.18) that implies in most natural cases that no-randomness-from-nothing holds for Schnorr randomness.

Observe that Proposition 7.2.9 implies that the measure μ_f is computable. Proposition 7.2.9 holds not only for computable functions, but for the larger class of effectively μ-approximable functions. However, Theorems 7.3.9 and 7.3.10 cannot hold for those functions: if f is in that class and x is Martin-Löf μ-random then $f(x)$ can be anything, unless $\mu(\{x\}) > 0$ (f remains in the class when changing it at x only). Here we see that one needs a notion of effectively measurable function that is well behaved on algorithmically random points. Such a notion exists and is called *layerwise computability*, presented in Section 7.4.5.

To give a specific application of these results, let us introduce the notion of a random closed set.

Example 7.3.11 (Random closed sets). In the context of computability theory, Barmpalias, Brodhead, Cenzer, Dashti and Weber [11] introduced the following definition of a "(Martin-Löf) random closed set". Every infinite tree T with no dead ends on $\{0,1\}^*$ can be represented by a ternary sequence X in $\{0,1,2\}^{\mathbb{N}}$ recursively as follows. Start at the root of T. If at the root T branches to both the left and the right, let $X(0) = 0$. If it only branches to the left, $X(0) = 1$ and if it only branches to the right, $X(0) = 2$. Recursively, for each node of the tree, similarly define a value of X to code how the tree branches at that node. If the sequence X is (Martin-Löf)

random, then the set $C \subseteq \{0,1\}^{\mathbb{N}}$ of paths through this tree is called a *(Martin-Löf) random closed set*.

The space $\mathscr{K}(2^{\mathbb{N}})$ of non-empty closed subsets of $2^{\mathbb{N}}$ forms a computable metric space with the Hausdorff metric (the topology is called the Fell topology). Moreover, the map $X \mapsto C$ which maps the representation for the tree T to its set of paths is computable as a map from $3^{\mathbb{N}}$ to $\mathscr{K}(2^{\mathbb{N}})$. If μ is the push-forward measure along this map, then our above theorems tell us that a closed set C is Martin-Löf random (in the specific sense that its encoding X is random) if and only if C is Martin-Löf μ-random. Actually, since this map is computably invertible, this result will hold for all of the major randomness notions listed in Section 7.3.1.

While this last example may seem simple, results such as these are key to working with random structures in mathematics. In many cases, they can turn a long proof into a short one. Here is an example.

Example 7.3.12 (Brownian motion). Consider a one-dimensional Martin-Löf random Brownian motion $B : [0,1] \to \mathbb{R}$. It is well known that the push-forward of the map $B \mapsto B(1)$ induces the Gaussian measure on $\mathbb{R}$. Therefore, it is natural to suspect that for every random Brownian motion B, the value of $B(1)$ is random (for the Gaussian measure) and if a is (Gaussian) random then there is a random Brownian motion such that $B(1) = a$. Indeed, for Martin-Löf randomness, the first fact follows from randomness preservation and the second from no-randomness-from-nothing. Observe that the Gaussian measure and the uniform measure on $\mathbb{R}$ have the same Martin-Löf random points (we will see why in Example 7.4.7).

We will see in Section 7.4.5 other randomness preservation results where Theorems 7.3.9 and 7.3.10 cannot be applied because the function involved is not continuous, but only measurable.

7.3.4 Product Spaces

It is well known that if one takes two *independent* sequences of independent identically distributed (i.i.d.) fair-coin flips $A = (a_0, a_1, a_2, \ldots)$ and $B = (b_0, b_1, b_2, \ldots)$ and interleaves them, then the resulting sequence $A \oplus B = (a_0, b_0, a_1, b_1, \ldots)$ is also an i.i.d. sequence of fair-coin flips. The concept of independence is essential to probability theory, and its analogue in algorithmic randomness is relative randomness.

Definition 7.3.13 (Relative Martin-Löf randomness). Let (X, μ) be a computable probability space and Y a computable metric space. A *uniform Martin-Löf test* is a computable sequence of effectively open sets $(U_n)_{n\in\mathbb{N}}$ on the product space $X \times Y$ such that for every $y \in Y$, we have $\mu\{x \in X : (x,y) \in U_n\} \leq 2^{-n}$. Say that $x \in X$ is *Martin-Löf random relative to* y if (x,y) is not contained in $\bigcap_n U_n$.

Theorem 7.3.14 (Van Lambalgen [80]). *Endow $\{0,1\}^{\mathbb{N}}$ with the fair-coin measure. For A and B in $\{0,1\}^{\mathbb{N}}$ the following are equivalent.*

1. *$A \oplus B$ is Martin-Löf random,*
2. *A is Martin-Löf random and B is Martin-Löf random relative to A.*

This result also extends to any computable product measure. Given two probability spaces (X,μ) and (Y,ν), the product measure $\mu \otimes \nu$ is the probability measure on $X \times Y$ given by $(\mu \otimes \nu)(A \times B) = \mu(A) \cdot \nu(B)$. The product measure operation is computable and we have the following version of van Lambalgen's theorem.

Theorem 7.3.15 (Van Lambalgen for product measures). *Let (X,μ) and (Y,ν) be two computable probability spaces. Let $x \in X$ and $y \in Y$. The following are equivalent.*

1. *(x,y) is Martin-Löf $(\mu \otimes \nu)$-random,*
2. *x is Martin-Löf μ-random and y is Martin-Löf ν-random relative to x.*

Finally, one may consider the case of an arbitrary probability measure on a product space. We saw in Section 7.2.3.1 how a probability measure π over $X \times Y$ can be decomposed into its marginal measure π_X over X, defined by $\pi_X(A) = \pi(A \times Y)$, and conditional measures $\pi(\cdots \mid x)$. We saw that the mapping $x \mapsto \pi(\cdot \mid x)$ is not always computable or effectively π_X-approximable. However, when it is computable one can prove a version of van Lambalgen's theorem for conditional probabilities.

Definition 7.3.16. Let X be a computable metric space. A *uniform Martin-Löf test* is a computable sequence of effectively open sets $(U_n)_n \in \mathbb{N}$ on the product space $\mathscr{M}_1(X) \times X$ such that for every $\mu \in \mathscr{M}_1(X)$, we have $\mu\{x \in X : (\mu,x) \in U_n\} \leq 2^{-n}$. Say that $x \in X$ is *Martin-Löf μ-random* if (μ,x) is not contained in $\bigcap_n U_n$.

One can naturally combine Definitions 7.3.13 and 7.3.16 to define Martin-Löf randomness for a non-computable measure relative to a non-computable oracle. A version of van Lambalgen's theorem for conditional probabilities was proved by Takahashi [109] on the Cantor space and holds on all computable metric spaces.

Theorem 7.3.17 (Van Lambalgen's theorem for conditional probabilities). *Let X,Y be computable metric spaces and π a computable measure over $X \times Y$. Assume that $x \mapsto \pi(\cdot \mid x)$ is a computable function. The following are equivalent.*

1. *(x,y) is Martin-Löf π-random,*
2. *x is Martin-Löf π_X-random and y is Martin-Löf $\pi(\cdot \mid x)$-random relative to x.*

In this result, the measure $\pi(\cdot \mid x)$ is computable relative to x, *for all $x \in X$ and uniformly in x*. Takahashi [110] extended this theorem by proving that the equivalence holds for a single x as long as $\pi(\cdot \mid x)$ is computable relative to x (while $\pi(\cdot \mid x')$ may not be computable relative to x' for $x' \neq x$), still assuming that the measure π is computable. Bauwens [12] showed that in general the equivalence in Theorem 7.3.17 fails when $\pi(\cdot \mid x)$ is not computable relative to x.

Now, let us turn to maps and random variables. Let (X,μ) be a computable probability space and Y a computable metric space. Let $T\colon X \to Y$ be a computable map. The push-forward measure μ_T is computable. Let $\mu(\cdot \mid T = y)$ denote the

conditional probability of x given that $T(x) = y$. Even if $\mu_T(y) = 0$ for all y, the conditional probability is well defined as a measurable function $y \mapsto \mu(\cdot \mid T = y)$. In particular, $y \mapsto \mu(\cdot \mid T = y)$ is the μ_T-almost everywhere unique function satisfying

$$\mu(A \cap T^{-1}(B)) = \int_B \mu(A \mid T = y)\, \mathrm{d}\mu_T(y)$$

for all measurable sets $A \subseteq X$ and $B \subseteq Y$.

Theorem 7.3.18 (Van Lambalgen's theorem for maps). *Let $T\colon (X, \mu) \to Y$ be as above and assume the conditional probability map $y \mapsto \mu(\cdot \mid T = y)$ is computable. Then the following are equivalent.*

1. *x is Martin-Löf μ-random,*
2. *$y := T(x)$ is Martin-Löf μ_T-random and x is Martin-Löf random w.r.t. $\mu(\cdot \mid T = y)$, relative to y.*

Proof. We simply apply the previous theorem to the measure π over $X \times Y$ defined as the push-forward of μ along the map $x \mapsto (x, T(x))$. That is, π is the measure supported on the graph of T whose marginal measure is μ. Conditioning π on x is given by $\pi(\cdot \mid x) = \delta_{T(x)}$ and is computable, so (x, y) is π-random iff condition 1. is satisfied (as $T(x)$ is always $\delta_{T(x)}$-random). Conditioning π on y is given by $\pi(\cdot \mid y) = \mu(\cdot \mid T = y)$ and is computable by assumption, so (x, y) is π-random iff condition 2. is satisfied. □

The abovementioned results can be extended in two natural ways. First, all of these versions of Van Lambalgen's theorem hold for Schnorr randomness under the correct notion of "relative Schnorr randomness". See Miyabe and Rute [90] and Rute [104] for details. Second, any theorem requiring that a map is "computable" can be extended to a layerwise computable map (or in the case of Schnorr randomness, a Schnorr layerwise computable map); see Definition 7.4.3. While the details are more technical, this allows for a much more natural setting.

Example 7.3.19 (Brownian motion again). We have seen in Example 7.3.12 that the values $B(1)$ taken by Martin-Löf random Brownian paths B at time 1 are exactly the Martin-Löf random reals w.r.t. the Gaussian measure, which coincide with the Martin-Löf random reals w.r.t. the Lebesgue measure, as explained in Example 7.4.7. The same result holds when replacing Martin-Löf randomness with Schnorr randomness on both sides. Indeed, the conditional probability of the map $B \mapsto B(1)$ conditioned on $B(1) = a$ is known as a *Brownian bridge landing at a*. Such objects are well studied in probability theory and the conditional probability map is computable. Therefore, for Schnorr randomness, the result follows from Van Lambalgen's theorem for maps (see Rute [104] for the details of this result).

7.4 Pointwise Computable Measure Theory

Let us take a step back regarding all the material developed so far. Computable measure theory is somewhat separated from other branches of computable analysis in several aspects:

- Computable analysis can usually be seen as computable topology. Indeed, the core concept of computable function in computable analysis is very close to the notion of continuous function, therefore it is appropriate on topological spaces but not on measure spaces.
- In computable analysis, the fundamental notion of computable function is defined in terms of a function mapping names of points to names of their images. However, the convenient notion of effectively measurable function does not comply with this definition.
- Computable analysis is generally about computing points or functions between points, however individual points are completely ignored in the definitions of effectively approximable and effectively measurable sets and functions.

We also saw that there is no unique way of investigating convergence theorems in terms of computability and that computable measure theory and algorithmic randomness give quite different insights on this problem. So it seems that we need to reconcile computable measure theory with computable topology and to investigate more precisely the relationship between the two parts of this chapter, computable measure theory and algorithmic randomness. Let us draw inspiration from the following text written by Doob in his book *Measure Theory* ([31] p. 101), comparing analysis before and after the advent of measure theory:

> In many contexts, measure theory widened the class of admissible domains and functions to the classes of measurable sets and measurable functions, and in so doing made it possible to apply the usual limiting procedures without leaving admissible classes. What was unexpected was that, in a reasonable sense, most of the old concepts were very nearly still present. Egorov's theorem showed that uniform convergence was nearly present whenever there was convergence. Lusin's theorem showed that the new measurable functions were nearly continuous. On the other hand, measure theory could be applied in abstract contexts where topology was inappropriate.

This phenomenon has consequences in computable measure theory. We will see that the "old concepts" from computable analysis can be used in computable measure theory, thanks to algorithmic randomness and more particularly Martin-Löf randomness.

This is possible because algorithmic randomness is inherently a pointwise approach to computable probability theory. We will see that many computable versions of results and constructs in measure and probability theory have a pointwise formulation. We have already seen that independence of random variables has a pointwise formulation, namely Van Lambalgen's theorem. We will see that effective absolute continuity of measures has a formulation in terms of randomness preservation, effective Egorov's theorem can be formulated in terms of uniform convergence on

random points, effective Lusin's theorem can be expressed as a form of uniform computability on random points, called layerwise computability, and so on.

The results in this section heavily rely on the important notion of randomness deficiency.

7.4.0.1 Randomness Deficiency

One of the first main results about Martin-Löf randomness is the existence of a universal Martin-Löf test, or a greatest Martin-Löf null set. More precisely there exists a Martin-Löf test $(U_n)_{n\in\mathbb{N}}$ such that for every Martin-Löf test $(V_n)_{n\in\mathbb{N}}$ there exists $c \in \mathbb{N}$ such that $V_{n+c} \subseteq U_n$ for all n. We fix such a universal test and define $\mathsf{ML}_n = X \setminus U_n$. The set of Martin-Löf random points can then be decomposed into levels $\mathsf{ML} = \bigcup_n \mathsf{ML}_n$ with $\mathsf{ML}_n \subseteq \mathsf{ML}_{n+1}$. If x is Martin-Löf random then the minimal n such that $x \in \mathsf{ML}_n$ is called the *randomness deficiency* of x. There are actually many equivalent ways of tests for Martin-Löf randomness. In each case there exists a universal test and its associated randomness deficiency notion. While these different quantities are not equal, they are computably related and all the results in this section remain true for these other notions.

The levels have large measures: $\mu(\mathsf{ML}_n) \geq 1 - 2^{-n}$ and are effectively closed sets, or Π^0_1-sets. One can prove more, as we now show.

7.4.1 Effective Tightness

In a complete separable metric space, every Borel probability measure is tight, i.e. assigns most of the weight to compact sets. This theorem is effective and witnessed by the Martin-Löf random points and their deficiencies.

Proposition 7.4.1 (Effective tightness). *Let X be a complete computable metric space and μ a computable Borel probability measure over X. The sets $\mathsf{ML}_n(\mu)$ are effectively compact, uniformly in n.*

This result appeared in [67]. More generally, Martin-Löf random points witness an effective version of Prokhorov's theorem: if $\mathscr{C} \subseteq \mathscr{M}_1(X)$ is an effectively compact class of Borel probability measures then $\mathsf{ML}_n(\mathscr{C}) := \bigcup_{\mu\in\mathscr{C}} \mathsf{ML}_n(\mu)$ is effectively compact [16]. For this, one needs to define Martin-Löf randomness for non-computable measures (see Definition 7.3.16 above and [49, 68]).

Proposition 7.4.1 is fundamental as it enables one to apply "old concepts" from computable analysis, involving effective compactness, to computable measure theory. We will see for instance how the effectiveness of the Pólya urn (Example 7.2.4) can easily be proved by a compactness argument rather than by probabilistic estimates (see Example 7.4.14).

This result is useful in the study of Martin-Löf randomness of Brownian motion, where the underlying space $\mathscr{C}([0,1],\mathbb{R}^d)$ is not compact, but the levels of Martin-

Löf random paths are effectively compact. For instance, it implies that for any computable Borel probability measure over $\mathscr{C}([0,1],\mathbb{R})$, the Martin-Löf random functions all have a computable modulus of uniform continuity. For the particular case of the Wiener measure, an explicit formula is given by Lévy's modulus of continuity theorem, but the computability of the modulus holds for any computable measure.

7.4.2 Effective Egorov's Theorem

If a point x is Martin-Löf random then it has a finite randomness deficiency, which cannot usually be computed or even bounded from an access to x. Moreover, having an upper bound on this deficiency gives important information about x that usually cannot be recovered from x. For instance, Davie [27] showed how this additional information can be used to compute the speed of convergence in the strong law of large numbers or the law of the iterated logarithm, or to bound the number of events in the Borel-Cantelli lemma. More generally, one can prove an effective version of Egorov's theorem, again involving Martin-Löf random points and their deficiencies [67].

Proposition 7.4.2 (Effective Egorov's theorem). *Let $f_n, f : X \to \mathbb{R}$ be uniformly computable functions. The following statements are equivalent:*

1. *f_n converge effectively μ-almost surely to f,*
2. *f_n converge effectively uniformly to f on each $\mathsf{ML}_k(\mu)$, uniformly in k.*

The second item means that given $k, \varepsilon > 0$, one can compute n such that

$$\sup_{x \in \mathsf{ML}_k(\mu)} |f_p(x) - f(x)| < \varepsilon$$

for all $p \geq n$.

Incidentally, this result relates more precisely the effective convergence notions investigated in the previous two parts of this chapter: effective almost sure convergence (Definition 7.2.17) and convergence on algorithmically random points. We saw that the former implies the latter (even for Schnorr random points) but the converse does not usually hold, and Proposition 7.4.2 shows the precise relationship.

7.4.3 Effective Lusin Theorem

We have just seen in Proposition 7.4.2 that if f_n are computable and converge effectively almost surely to f then for an individual x, one can compute the speed of convergence of $f_n(x)$ to $f(x)$ simply from any upper bound on its randomness deficiency. In particular, one can compute $f(x)$ from x and such an upper bound. We make this observation into a definition [68].

Let (Y, δ_Y) be a set with a representation.

Definition 7.4.3. A function $f : X \to Y$ is *μ-layerwise computable* if for all $k \in \mathbb{N}$ and all $x \in \mathsf{ML}_k(\mu)$, $f(x)$ is computable relative to x, uniformly in k, x.

This means that there is a type-two Turing machine taking as inputs $k \in \mathbb{N}$ and a name of $x \in \mathsf{ML}_k(\mu)$ and outputs a name of $f(x)$.

The next result [66] shows that this notion is an alternative to the notions of effectively approximable and effectively measurable functions. It is at the same time an effective version of Lusin's theorem, which states that every measurable function is *nearly continuous*, i.e. continuous on a compact subset of measure arbitrarily close to 1.

Proposition 7.4.4 (Effective Lusin theorem). *Let $f : X \to Y$. The following statements are equivalent:*

1. *f is effectively μ-approximable,*
2. *f coincides μ-almost everywhere with a μ-layerwise computable function.*

Note that a μ-layerwise computable function is computable hence continuous on each $\mathsf{ML}_k(\mu)$, so this proposition is indeed an effective version of Lusin's theorem.

There are good reasons to use the notion of layerwise computable function in place of effectively measurable/approximable function. On the one hand it is much better adapted to the study of Martin-Löf randomness. For instance, two layerwise computable functions that coincide almost everywhere actually coincide on the Martin-Löf random points. Almost all the theorems about Martin-Löf random points involving computable functions actually hold for layerwise computable functions, thus this notion is a suitable notion of effectively measurable function that is well behaved w.r.t. Martin-Löf randomness. On the other hand it enables one to use "old concepts" from computable analysis in computable measure theory, because they are instances of computable functions between represented spaces for a suitable representation (a random point is represented by a name and an upper bound on its randomness deficiency), and they are almost like computable functions, with some non-uniformity.

7.4.4 Effective Absolute Continuity

We have already seen randomness preservation theorems, which are theorems about preservation of effective null sets. The question of the preservation of null sets is naturally raised in another situation, when comparing two measures. We recall that a measure ν is absolutely continuous w.r.t. a measure μ, written $\nu \ll \mu$, if every μ-null set is also ν-null. It is natural to investigate effective versions, where null sets are replaced by any notion of *effective* null sets. For a probability measure μ, let $\mathsf{Sch}(\mu)$, $\mathsf{CR}(\mu)$ and $\mathsf{Ku}(\mu)$ denote the sets of Schnorr random, computably random, and Kurtz random points respectively.

Theorem 7.4.5 (Bienvenu, Merkle [17]). *The following implications hold and are tight:*

$$\left.\begin{array}{r} \mathsf{CR}(\nu) \subseteq \mathsf{CR}(\mu) \Longrightarrow \mathsf{ML}(\nu) \subseteq \mathsf{ML}(\mu) \\ \mathsf{Sch}(\nu) \subseteq \mathsf{Sch}(\mu) \end{array}\right\} \Longrightarrow \nu \ll \mu \Longrightarrow \mathsf{Ku}(\nu) \subseteq \mathsf{Ku}(\mu).$$

The implication $\mathsf{ML}(\nu) \subseteq \mathsf{ML}(\mu) \implies \nu \ll \mu$ was independently proved by Archibald, Brattka and Heuberger [4].

We have seen another effective version of absolute continuity in Definition 7.2.14. It is related to algorithmic randomness through a characterization that involves randomness deficiency.

Proposition 7.4.6. *The following statements are equivalent:*

1. *ν is effectively absolutely continuous w.r.t. μ,*
2. *There exists a computable function $\psi : \mathbb{N} \to \mathbb{N}$ such that $\mathsf{ML}_n(\nu) \subseteq \mathsf{ML}_{\psi(n)}(\mu)$.*

Proof. 1. witnessed by φ implies 2. with $\psi(n) = \varphi(n+c)$ for some c. Indeed, as $\mu(\mathsf{ML}_{\varphi(n)}(\mu)) \geq 1 - 2^{-\varphi(n)}$, one has $\nu(\mathsf{ML}_{\varphi(n)}(\mu)) \geq 2^{-n}$ so there exists a constant $c \in \mathbb{N}$ such that $\mathsf{ML}_n(\nu) \subseteq \mathsf{ML}_{\varphi(n+c)}(\mu)$ for all n.

2. implies 1. One can prove that if $\mathsf{ML}_n(\nu) \subseteq \mathsf{ML}_{\psi(n)}(\mu)$ for all n then ν is effectively absolutely continuous w.r.t. μ with $\varphi(n) = \psi(n) + c$ for some $c \in \mathbb{N}$. The argument is a refinement of the proof of Theorem 7.4.5 (Proposition 3.3 in [17]). □

In particular, if f is $L^1(X, \mu)$ computable and ν is the measure with density f then ν is effectively absolutely continuous w.r.t. μ by Proposition 7.2.15 so every Martin-Löf μ-random point is also Martin-Löf ν-random. This is also true for Schnorr randomness.

Example 7.4.7 (Brownian motion continued). We have seen in Example 7.3.12 that the values taken by Martin-Löf random paths $B : [0,1] \to \mathbb{R}$ at time 1 are exactly the reals that are Martin-Löf random w.r.t. the Gaussian measure over $\mathbb{R}$. The Gaussian measure is equivalent to the Lebesgue measure, in the sense that they are both absolutely continuous w.r.t. one another, moreover they are *effectively* absolutely continuous w.r.t. one another. As a result they have the same Martin-Löf random points. All in all, the set of values $B(1)$ for Martin-Löf random paths B is exactly the set of Martin-Löf random reals w.r.t. the Lebesgue measure. The remark following Theorem 7.3.18 implies that the same result holds for Schnorr randomness.

7.4.5 Properties of Layerwise Computable Functions

It turns out that "naturally defined" effectively measurable functions are usually already layerwise computable. For instance, the fat Cantor set (Example 7.2.6), or any effectively closed set or effectively open set whose measure is computable is

not only effectively approximable (Proposition 7.2.7), but actually layerwise computable. More generally, the effectively measurable sets in the sense of Edalat [34] coincide with the layerwise computable sets. So layerwise computability is a stronger notion that is in practice a handy substitute for effective approximability, as it has stronger properties and behaves well w.r.t. Martin-Löf randomness.

Moreover layerwise computable functions are instances of computable functions between represented spaces (for a suitable representation), contrary to effectively measurable or approximable functions. Therefore, they mostly behave as usual computable functions, with essentially the same proofs. We list a few simple closure properties of the class of layerwise computable functions:

- If $f, g : X \to \mathbb{R}$ are μ-layerwise computable then so are $f+g, f-g, fg, |f|$, etc.
- If $f_n : X \to [0,1]$ are uniformly μ-layerwise computable then so is $\sum_n 2^{-n} f_n$,
- If $f_n : X \to \mathbb{R}$ are uniformly μ-layerwise computable and converge effectively μ-almost surely, then their pointwise limit is μ-layerwise computable,
- If Y is another computable metric space and $f : X \to Y$ is μ-layerwise computable and one-to-one then f^{-1} is μ_f-layerwise computable.

The original goal of layerwise computability was to have an effective notion of measurable function that behaves well on Martin-Löf random points. This is indeed the case, as most of the theorems about Martin-Löf random points involving computable functions also hold for layerwise computable functions, with essentially the same proof. For instance, in Birkhoff's ergodic theorem for Martin-Löf random points (Theorem 7.3.3), the functions T, f can be assumed to be μ-layerwise computable only [67].

We have already seen that computable functions preserve randomness (Theorems 7.3.9 and 7.3.10), which has interesting consequences (Examples 7.3.11 and 7.3.12). These results extend to the larger class of layerwise computable functions.

Theorem 7.4.8 (Hoyrup, Rojas [67]). *Let $f : X \to Y$ be μ-layerwise computable.*

- *The push-forward measure $\nu = \mu_f = \mu(f^{-1}(.))$ is computable, and*

$$f(\mathsf{ML}(\mu)) = \mathsf{ML}(\nu). \tag{7.7}$$

- *If $g : Y \to Z$ is ν-layerwise computable then $g \circ f$ is μ-layerwise computable.*
- *If f is moreover one-to-one then $f^{-1} : Y \to X$ is ν-layerwise computable.*

Proof. The sets $\mathsf{ML}_k(\mu)$ are effectively compact, so their images $f(\mathsf{ML}_k(\mu))$ are also effectively compact, thus their complements $V_k = Y \setminus f(\mathsf{ML}_k(\mu))$ are effectively open. By definition of μ_f, one has $\mu_f(f(\mathsf{ML}_k(\mu))) = \mu(f^{-1}(f(\mathsf{ML}_k(\mu)))) \geq \mu(\mathsf{ML}_k(\mu)) \geq 1 - 2^{-k}$. As a result, $\mu_f(V_k) \leq 2^{-k}$ so $(V_k)_{k\in\mathbb{N}}$ is a Martin-Löf μ_f-test. If y is Martin-Löf μ_f-random then $y \notin V_k$ for some k, which means that $y = f(x)$ for some $x \in \mathsf{ML}_k(\mu)$.

We give the proof of the last item, which is again a simple compactness argument based on Proposition 7.4.1. If a continuous function defined on a compact set is one-to-one, then its inverse is continuous. This is effective: if a computable function defined on an effectively compact set is one-to-one then its inverse is computable,

and this is uniform. The result is just a direct application of that result on each level $\mathsf{ML}_n(\mu)$. □

The proof makes essential use of the effective compactness of the sets $\mathsf{ML}_k(\mu)$ (Proposition 7.4.1).

Equality (7.7) is the version of Theorems 7.3.9 and 7.3.10 for measurable functions (it is not true for effectively approximable/measurable functions). Observe that in this case, an upper bound on the randomness deficiency of $f(x)$ can easily be computed from an upper bound on the randomness deficiency of x, as $f(\mathsf{ML}_k(\mu)) \subseteq \mathsf{ML}_{k+c}(\nu)$ for some constant $c \in \mathbb{N}$ and all $k \in \mathbb{N}$.

The last item calls for a few remarks. The class of layerwise computable functions is closed under taking the inverse, so by Proposition 7.4.4, the same closure property holds for effectively approximable functions, while a direct proof would not be as simple. That last item also implies in particular that while the inverse of a computable one-to-one function is not computable in general, it is always layerwise computable.

7.4.6 Randomness via Encoding

Theorem 7.4.8 has an important application that we present now.

Originally Martin-Löf introduced his notion of randomness for infinite binary sequences [85]. In the subsequent literature, there have been mainly two ways of defining algorithmic randomness for other classes of objects. One way is to extend Martin-Löf's definition to other topological spaces and directly apply Definition 7.3.1 [58, 49, 68]. The other is to encode objects into more primitive objects like infinite binary sequences and then declare an object to be random if its code is random. It often happens that these two approaches induce the same notion of randomness. Moreover, when several non-equivalent encodings are possible, they often turn out to give the same class of random objects.

In this section we give two results that explain this phenomenon, and are applications of Theorem 7.4.8. These results intuitively show that several representations of objects which are non-computably equivalent are often computably equivalent on the random objects, if they induce the same computable measure.

First observe that the algorithmic notions of randomness induced by an encoding depend on two factors: the induced or push-forward measure, and the computability properties of the encoding. If two encodings induce non-equivalent measures (measures that are not absolutely continuous w.r.t. each other), then they cannot induce the same class of random objects, essentially by Theorem 7.4.5. For instance, the signed-digit representation of real numbers induces a notion of randomness that is disjoint from randomness w.r.t. the Lebesgue measure [4]. So we will assume that the encodings induce the same measure.

7.4.6.1 Semicomputable Representations

Usually the computability of the push-forward measure does not imply much about the function, except for restricted classes of functions.

Proposition 7.4.9. *Let $f : X \to \mathbb{R}$ be lower semicomputable. f is μ-layerwise computable if and only the push-forward μ_f is computable.*

Proof. We already saw that effectively μ-approximable, hence μ-layerwise computable functions, have a computable push-forward measure.

In the other direction, if μ_f is computable then there is a computable dense sequence of real numbers $(r_i)_{i\in\mathbb{N}}$ such that $\mu_f\{r_i\} = 0$ for all i. For each i, the set $f^{-1}(r_i, +\infty)$ is effectively open and $\mu(f^{-1}(r_i, +\infty)) = \mu_f(r_i, +\infty)$ is computable, so it is a μ-layerwise computable set. In other words, given $n \in \mathbb{N}$ and $x \in \mathsf{ML}_n(\mu)$, one can decide for each i whether $f(x) > r_i$, which enables one to compute $f(x)$. □

The same argument can be applied to "semicomputable" functions into spaces other than $\mathbb{R}$.

For instance if $f : X \to \{0,1\}^{\mathbb{N}}$ is such that there is a machine enumerating $f(x)$ (identified with a subset of $\mathbb{N}$) given x, and the push-forward μ_f is computable (over the Cantor space as a computable metric space), then f is μ-layerwise computable: for each n, the set $\{x \in X : f(x) \text{ has a 1 at position } n\}$ is effectively open and has computable measure so it is μ-layerwise computable, uniformly in n.

Example 7.4.10 (Random closed sets continued). We have already seen two equivalent ways of defining Martin-Löf random closed subsets of the Cantor space, one directly on the space of closed sets with the Hausdorff metric, the other by encoding closed sets as binary trees without dead ends. There is yet another equivalent one defined in [75] by generating a random tree by a Galton-Watson process as follows. Start with the root node. For each node $w \in \{0,1\}^*$ added to the tree, independently add its extensions $w0$ and $w1$ to the tree, each with probability $2/3$. This tree does have dead ends and can be encoded as a binary sequence. The function mapping such a sequence to the closed set of infinite branches of the tree is not computable, because one can never be sure that a node will have an infinite extension.

However this map is "semicomputable" in the sense that given a binary sequence encoding a closed set, one can enumerate the cylinders that are disjoint from the closed set. It happens that the push-forward of this map is a computable measure, which is almost the same measure as the one from Example 7.3.11 (the difference is that it gives positive weight to the empty set). By the argument of Proposition 7.4.9, this map is then layerwise computable, and the closed sets coded by Martin-Löf random binary sequences are exactly the empty set and the Martin-Löf random closed sets from Example 7.3.11. The equivalence was proved in [30] and [9].

7.4.6.2 Metric Representations

Here we show that different metrics often induce the same notion of Martin-Löf random point and give the same information for those points. More precisely, we

consider two metrics d, d' such that d' is weaker than d in the sense that the identity $\mathrm{id} : (X,d) \to (X,d')$ is computable (equivalently, $d' : (X,d) \times (X,d) \to \mathbb{R}$ is computable). Observe that its inverse $\mathrm{id} : (X,d') \to (X,d)$ is not in general computable unless (X,d) is effectively compact, which we do not assume here.

Proposition 7.4.11. *Let (X,d) be a complete computable metric space endowed with a computable probability measure μ. Let d' be a metric over X such that $d' : X \times X \to \mathbb{R}$ is computable. The Martin-Löf μ-random points are the same in the spaces (X,d) and (X,d'). Moreover, the identity from (X,d') to (X,d) is μ-layerwise computable.*

Proof. The identity from (X,d) to (X,d') is computable, maps μ to μ and is one-to-one, so by Theorems 7.3.9 and 7.3.10 it maps the set of Martin-Löf μ-random points in (X,d) exactly to the set of Martin-Löf μ-random points in (X,d'), and its inverse is μ-layerwise computable by Theorem 7.4.8. □

Example 7.4.12 (Representations of Brownian motion). We have already met the Martin-Löf random elements of the computable metric space $(\mathscr{C}([0,1]), \|\cdot\|_\infty)$ endowed with the Wiener measure. Another way of defining Martin-Löf random paths considered in [37] is to code the values of a continuous function $B : [0,1] \to \mathbb{R}$ at the rational numbers into a binary sequence and considering a particular measure over the Cantor space, and then say that a path is random if its encoding is random. It is proved in [37] that the class of random paths is the same, and in [28] that the function mapping an encoding to the function, although not computable, is layerwise computable.

This result is a direct application of Proposition 7.4.11, by considering the weaker metric $d'(f,g) = \sum_{i \in \mathbb{N}} 2^{-i} |f(q_i) - g(q_i)|$ where $(q_i)_{i \in \mathbb{N}}$ is a computable enumeration of the rational numbers. Proposition 7.4.11 tells us that the Martin-Löf random paths are the same using the uniform distance and the metric d', and that the values of f on the rationals are sufficient to compute f at any real number, given an upper bound on the randomness deficiency of f. Indeed, the randomness deficiency of f automatically gives a modulus of uniform continuity of f (though the proof of Proposition 7.4.11 is more abstract).

7.4.7 Recovering a Distribution from a Sample

We now present concrete examples where layerwise computability is the right substitute for computability, in the absence of continuity.

According to Birkhoff's ergodic theorem, for a measurable function $T : X \to X$ and an invariant measure μ, for μ-almost every x the orbit of x under T has a limit distribution μ_x in X. This means that for those x, for all "simple" sets A (sets in a fixed countable family), one has

$$\mu_x(A) = \lim_{n \to \infty} \frac{1}{n} \sum_{i=0}^{n-1} \mathbf{1}_A \circ T^i(x).$$

What about the computability of the measure μ_x? Can it be computed given x as oracle? Usually the function $x \mapsto \mu_x$ is discontinuous so it cannot be computable. It turns out that in many cases it is μ-layerwise computable, which adds another equivalence to Proposition 7.2.24.

Proposition 7.4.13 (Hoyrup [62]). *Let X be a complete computable metric space, μ a computable Borel probability measure and $T : X \to X$ a μ-layerwise computable function. The following statements are equivalent:*

- *The ergodic decomposition m of μ is computable,*
- *The mapping $x \mapsto \mu_x$ is μ-layerwise computable.*

In that case, $\mathsf{ML}(\mu) = \bigcup_{\nu \in \mathsf{ML}(m)} \mathsf{ML}(\nu)$ and for $x \in \mathsf{ML}(\mu)$, $\mu_x \in \mathsf{ML}(m)$ is ergodic and $x \in \mathsf{ML}(\mu_x)$.

The proof of the direct implication heavily relies on the effective compactness of the levels of Martin-Löf random points.

Example 7.4.14 (Pólya urn again). We already saw in Example 7.2.27 that the class of Bernoulli measures over $\{0,1\}^{\mathbb{N}}$ is effectively compact, which implies that the decomposition of the measure μ into Bernoulli measures is computable. It also implies by Proposition 7.4.13 that the mapping $x \mapsto \mu_x$ is μ-layerwise computable. But this mapping is essentially the function $p(x)$ from Example 7.2.4, sending a random sequence x to the limit frequency of occurrences of 1 in x. This function is then not only effectively μ-approximable, but also μ-layerwise computable. It is worth observing that the proof neither relies on the particularities of μ nor on probability estimates for the speed of convergence, but rather on an abstract effective compactness argument.

Observe that in the case of Proposition 7.4.13, the ergodic measures are computable from their random points. The more general case of a measure which can be computed from its random points has been studied by Bienvenu and Monin [18], who obtained a characterization which gives an answer to the following question: which measures can be computed from all their Martin-Löf random points? When one requires that the computation of the measure is effective in the randomness deficiency of the random points, one gets a precise answer, as shown below.

We say that two measures are *effectively orthogonal* if they do not have Martin-Löf random points in common. For instance, in a dynamical system two distinct ergodic measures are pairwise effectively orthogonal.

Theorem 7.4.15 (Bienvenu, Monin [18]). *Let X be an effectively compact computable metric space and $\mathscr{C} \subseteq \mathscr{M}_1(X)$ a class of Borel probability measures. The following statements are equivalent:*

- *$\mathscr{C}$ is contained in an effectively compact class of pairwise effectively orthogonal measures,*
- *There is a total computable function $F : \mathbb{N} \times X \to \mathscr{M}_1(X)$ such that for every $\mu \in \mathscr{C}$, every $n \in \mathbb{N}$ and every $x \in \mathsf{ML}_n(\mu)$, $F(n,x) = \mu$.*

This implies in particular that the function that for every $\mu \in \mathscr{C}$ maps $x \in \mathsf{ML}(\mu)$ to μ is well defined and μ-layerwise computable for all $\mu \in \mathscr{C}$. The algorithm is moreover independent of μ and is defined everywhere.

Proof (outline). Assume that $\mathscr{C}$ is an effectively compact class of pairwise effectively orthogonal measures. If $x \in \mathsf{ML}_n(\mu)$ then $x \notin \mathsf{ML}_n(\nu)$ for all $\nu \neq \mu$ in $\mathscr{C}$ as ν and μ do not have common Martin-Löf random points. As a result, $\{\mu\} = \{\nu \in \mathscr{C} : x \in \mathsf{ML}_n(\nu)\}$, which is effectively compact relative to x as $\mathscr{C}$ is effectively compact, so μ is computable relative to x. The argument is uniform in x and n, which defines a computable function $F(n,x)$. By the computable Tietze extension theorem [117], $F(n,.)$ can be computably extended outside $\bigcup_{\mu\in\mathscr{C}} \mathsf{ML}_n(\mu)$, which is effectively compact.

Conversely, given $\mathscr{C}$ and F, let $\mathscr{K} = \{\mu : \forall n \in \mathbb{N}, \forall x \in \mathsf{ML}_n(\mu), F(n,x) = \mu\}$. One can show that $\mathscr{K}$ is an effectively closed class of effectively orthogonal measures containing $\mathscr{C}$. As X is effectively compact, so are $\mathscr{M}(X)$ and $\mathscr{K}$. Totality of F is essential here to make $\mathscr{K}$ effectively closed. □

Note that in Theorem 7.4.15 one requires the function F to be total. One can extend this result, with essentially the same argument, to the case when F is just defined on Martin-Löf random points, like layerwise computable functions.

Theorem 7.4.16. *Let X be an effectively compact computable metric space and $\mathscr{C} \subseteq \mathscr{M}_1(X)$ a class of Borel probability measures. The following statements are equivalent:*

- *$\mathscr{C}$ is contained in an effectively compact class $\mathscr{K}$ of measures such that every $\mu \in \mathscr{C}$ is effectively orthogonal with every $\nu \in \mathscr{K}$, $\nu \neq \mu$,*
- *There is a partial computable function $F :\subseteq \mathbb{N} \times X \to \mathscr{M}_1(X)$ such that for every $\mu \in \mathscr{C}$, every $n \in \mathbb{N}$ and every $x \in \mathsf{ML}_n(\mu)$, $F(n,x) = \mu$.*

The second condition exactly says that the function that for every $\mu \in \mathscr{C}$ maps $x \in \mathsf{ML}(\mu)$ to μ is well defined and μ-layerwise computable for all $\mu \in \mathscr{C}$, with an algorithm that is independent of μ.

References

1. Ackerman, N.L., Freer, C.E., Roy, D.M.: Noncomputable conditional distributions. In: 26th Annual IEEE Symposium on Logic in Computer Science—LICS 2011, pp. 107–116. IEEE Computer Soc., Los Alamitos, CA (2011)
2. Ackerman, N.L., Freer, C.E., Roy, D.M.: On computability and disintegration. Mathematical Structures in Computer Science **27**(8), 1287–1314 (2017)
3. Allen, K., Bienvenu, L., Slaman, T.A.: On zeros of Martin-Löf random Brownian motion. Journal of Logic and Analysis **6**, Paper 9, 34 (2014)
4. Archibald, M., Brattka, V., Heuberger, C.: Randomness with respect to the signed-digit representation. Fundamenta Informaticae **83**(1–2), 1–19 (2008)
5. Asarin, E., Pokrovskii, A.: Use of the Kolmogorov complexity in analyzing control system dynamics. Avtomatika i Telemekhanika **1** (1986)

6. Avigad, J.: Uncomputably noisy ergodic limits. Notre Dame Journal of Formal Logic **53**(3), 347–350 (2012)
7. Avigad, J., Dean, E.T., Rute, J.: Algorithmic randomness, reverse mathematics, and the dominated convergence theorem. Annals of Pure and Applied Logic **163**(12), 1854–1864 (2012)
8. Avigad, J., Gerhardy, P., Towsner, H.: Local stability of ergodic averages. Transactions of the American Mathematical Society **362**, 261–288 (2010)
9. Axon, L.M.: Algorithmically random closed sets and probability. Ph.D. thesis. Department of Mathematics, University of Notre Dame (2010)
10. Axon, L.M.: Martin-Löf randomness in spaces of closed sets. The Journal of Symbolic Logic **80**(2), 359–383 (2015)
11. Barmpalias, G., Brodhead, P., Cenzer, D., Dashti, S., Weber, R.: Algorithmic randomness of closed sets. Journal of Logic and Computation **17**(6), 1041–1062 (2007)
12. Bauwens, B.: Conditional measure and the violation of van Lambalgen's theorem for Martin-Löf randomness. Theory of Computing Systems **60**(2), 1–10 (2017)
13. Becher, V., Figueira, S.: An example of a computable absolutely normal number. Theoretical Computer Science **270**, 947–958 (2002)
14. Bienvenu, L., Day, A., Mezhirov, I., Shen, A.: Ergodic-type characterizations of algorithmic randomness. In: F. Ferreira, B. Löwe, E. Mayordomo, L.M. Gomes (eds.) CiE 2010, *Lecture Notes in Computer Science*, vol. 6158, pp. 49–58. Springer, Berlin (2010)
15. Bienvenu, L., Day, A.R., Hoyrup, M., Mezhirov, I., Shen, A.: A constructive version of Birkhoff's ergodic theorem for Martin-Löf random points. Information and Computation **210**, 21–30 (2012)
16. Bienvenu, L., Gács, P., Hoyrup, M., Rojas, C., Shen, A.: Algorithmic tests and randomness with respect to a class of measures. Proceedings of the Steklov Institute of Mathematics **274**(1), 34–89 (2011)
17. Bienvenu, L., Merkle, W.: Effective randomness for computable probability measures. In: D. Cenzer, R. Dillhage, T. Grubba, K. Weihrauch (eds.) Proceedings of the Third International Conference on Computability and Complexity in Analysis, *Electronic Notes in Theoretical Computer Science*, vol. 167, pp. 117–130. Elsevier, Amsterdam (2007), CCA 2006, Gainesville, Florida, USA, November 1–5, 2006
18. Bienvenu, L., Monin, B.: Von Neumann's biased coin revisited. In: Proceedings of the 27th Annual IEEE Symposium on Logic in Computer Science, LICS 2012, Dubrovnik, Croatia, June 25-28, 2012, pp. 145–154. IEEE Computer Society (2012)
19. Bishop, E.: Foundations of Constructive Analysis. McGraw-Hill, New York (1967)
20. Bosserhoff, V.: Are unbounded linear operators computable on the average for Gaussian measures? Journal of Complexity **24**(4), 477–491 (2008)
21. Bosserhoff, V.: Notions of probabilistic computability on represented spaces. Journal of Universal Computer Science **14**(6), 956–995 (2008)
22. Brattka, V., Gherardi, G., Hölzl, R.: Las Vegas computability and algorithmic randomness. In: E.W. Mayr, N. Ollinger (eds.) 32nd International Symposium on Theoretical Aspects of Computer Science (STACS 2015), *Leibniz International Proceedings in Informatics (LIPIcs)*, vol. 30, pp. 130–142. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2015)
23. Brattka, V., Gherardi, G., Hölzl, R.: Probabilistic computability and choice. Information and Computation **242**, 249–286 (2015)
24. Brattka, V., Miller, J.S., Nies, A.: Randomness and differentiability. Transactions of the American Mathematical Society **368**(1), 581–605 (2016)
25. Brin, M., Stuck, G.: Introduction to Dynamical Systems. Cambridge University Press (2002)
26. Brodhead, P., Cenzer, D., Remmel, J.B.: Random continuous functions. In: D. Cenzer, R. Dillhage, T. Grubba, K. Weihrauch (eds.) Proceedings of the Third International Conference on Computability and Complexity in Analysis, *Electronic Notes in Theoretical Computer Science*, vol. 167. Elsevier, Amsterdam (2007), CCA 2006, Gainesville, Florida, USA, November 1–5, 2006
27. Davie, G.: The Borel-Cantelli lemmas, probability laws and Kolmogorov complexity. Annals of Probability **29**(4), 1426–1434 (2001)

28. Davie, G., Fouché, W.L.: On the computability of a construction of Brownian motion. Mathematical Structures in Computer Science **23**, 1257–1265 (2013)
29. Demuth, O.: The differentiability of constructive functions of weakly bounded variation on pseudo numbers (Russian). Comment. Math. Univ. Carolinae **16**(3), 583–599 (1975)
30. Diamondstone, D., Kjos-Hanssen, B.: Members of random closed sets. In: K. Ambos-Spies, B. Löwe, W. Merkle (eds.) Mathematical Theory and Computational Practice, 5th Conference on Computability in Europe, CiE 2009, Heidelberg, Germany, July 19-24, 2009. Proceedings, *Lecture Notes in Computer Science*, vol. 5635, pp. 144–153. Springer (2009)
31. Doob, J.: Measure Theory. *Graduate Texts in Mathematics*. Springer-Verlag (1994)
32. Downey, R.G., Hirschfeldt, D.R.: Algorithmic Randomness and Complexity. *Theory and Applications of Computability*. Springer, New York (2010)
33. Edalat, A.: The Scott topology induces the weak topology. In: LICS '96: Proceedings of the 11th Annual IEEE Symposium on Logic in Computer Science, p. 372. IEEE Computer Society, Washington, DC, USA (1996)
34. Edalat, A.: A computable approach to measure and integration theory. Information and Computation **207**(5), 642–659 (2009)
35. Férée, H., Ziegler, M.: On the computational complexity of positive linear functionals on $C[0;1]$. In: I.S. Kotsireas, S.M. Rump, C.K. Yap (eds.) Mathematical Aspects of Computer and Information Sciences, *Lecture Notes in Computer Science*, vol. 9582, pp. 489–504. Springer, Cham (2016), 6th International Conference, MACIS 2015, Berlin, Germany, November 11–13, 2015, Revised Selected Papers
36. Fouché, W.: Arithmetical representation of Brownian motion I. The Journal of Symbolic Logic **65**, 421–442 (2000)
37. Fouché, W.: The descriptive complexity of Brownian motion. Advances in Mathematics **155**, 317–343 (2000)
38. Fouché, W.: Dynamics of a generic Brownian motion: Recursive aspects. Theoretical Computer Science **394**, 175–186 (2008)
39. Fouché, W.L.: Fractals generated by algorithmically random Brownian motion. In: K. Ambos-Spies, B. Löwe, W. Merkle (eds.) Mathematical Theory and Computational Practice, *Lecture Notes in Computer Science*, vol. 5635, pp. 208–217. Springer, Berlin (2009), 5th Conference on Computability in Europe, CiE 2009, Heidelberg, Germany, July 19-24, 2009. Proceedings
40. Fouché, W.L.: Diophantine properties of Brownian motion: recursive aspects. In: V. Brattka, H. Diener, D. Spreen (eds.) Logic, Computation, Hierarchies, *Ontos Mathematical Logic*, vol. 4, pp. 139–156. De Gruyter, Boston (2014)
41. Fouché, W.L.: Kolmogorov complexity and the geometry of Brownian motion. Mathematical Structures in Computer Science **25**(7), 1590–1606 (2015)
42. Fouché, W.L., Mukeru, S., Davie, G.: Fourier spectra of measures associated with algorithmically random Brownian motion. Logical Methods in Computer Science **10**(3), 3:20, 24 (2014)
43. Franklin, J.N.Y., Greenberg, N., Miller, J.S., Ng, K.M.: Martin-Löf random points satisfy Birkhoff's ergodic theorem for effectively closed sets. Proceedings of the American Mathematical Society **140**(10), 3623–3628 (2012)
44. Franklin, J.N.Y., Towsner, H.: Randomness and non-ergodic systems. Moscow Mathematical Journal **14**(4), 711–744, 827 (2014)
45. Freer, C., Kjos-Hanssen, B., Nies, A., Stephan, F.: Algorithmic aspects of Lipschitz functions. Computability **3**(1), 45–61 (2014)
46. Freer, C.E., Roy, D.M.: Computable exchangeable sequences have computable de Finetti measures. In: K. Ambos-Spies, B. Löwe, W. Merkle (eds.) Mathematical Theory and Computational Practice, *Lecture Notes in Computer Science*, vol. 5635, pp. 218–231. Springer, Berlin (2009)
47. Freer, C.E., Roy, D.M.: Computable de Finetti measures. Annals of Pure and Applied Logic **163**(5), 530–546 (2012)
48. Friedman, H.: The computational complexity of maximization and integration. Advances in Mathematics **53**, 80–98 (1984)

49. Gács, P.: Uniform test of algorithmic randomness over a general space. Theoretical Computer Science **341**, 91–137 (2005)
50. Gács, P., Hoyrup, M., Rojas, C.: Randomness on computable probability spaces—a dynamical point of view. In: S. Albers, J.Y. Marion (eds.) STACS 2009: 26th International Symposium on Theoretical Aspects of Computer Science, *LIPIcs. Leibniz Int. Proc. Inform.*, vol. 3, pp. 469–480. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern (2009)
51. Gács, P., Hoyrup, M., Rojas, C.: Randomness on computable probability spaces—a dynamical point of view. Theory of Computing Systems **48**(3), 465–485 (2011)
52. Galatolo, S., Hoyrup, M., Rojas, C.: A constructive Borel-Cantelli lemma. Constructing orbits with required statistical properties. Theoretical Computer Science **410**, 2207–2222 (2009)
53. Galatolo, S., Hoyrup, M., Rojas, C.: Computing the speed of convergence of ergodic averages and pseudorandom points in computable dynamical systems. In: X. Zheng, N. Zhong (eds.) Proceedings Seventh International Conference on Computability and Complexity in Analysis 2010, *Electronic Proceedings in Theoretical Computer Science*, vol. 24, pp. 7–18. Open Publishing Association (2010), Zhenjiang, China, 21-25th June 2010
54. Galatolo, S., Hoyrup, M., Rojas, C.: Effective symbolic dynamics, random points, statistical behavior, complexity and entropy. Information and Computation **208**(1), 23–41 (2010)
55. Galatolo, S., Hoyrup, M., Rojas, C.: Dynamics and abstract computability: computing invariant measures. Discrete and Continuous Dynamical Systems. Series A **29**(1), 193–212 (2011)
56. Galatolo, S., Hoyrup, M., Rojas, C.: Statistical properties of dynamical systems—simulation and abstract computation. Chaos, Solitons and Fractals **45**(1), 1–14 (2012)
57. Hertling, P., Wang, Y.: Invariance properties of random sequences. Journal of Universal Computer Science **3**(11), 1241–1249 (1997)
58. Hertling, P., Weihrauch, K.: Randomness spaces. In: K.G. Larsen, S. Skyum, G. Winskel (eds.) Automata, Languages and Programming, *Lecture Notes in Computer Science*, vol. 1443, pp. 796–807. Springer, Berlin (1998), 25th International Colloquium, ICALP'98, Aalborg, Denmark, July 1998
59. Hertling, P., Weihrauch, K.: Random elements in effective topological spaces with measure. Information and Computation **181**(1), 32–56 (2003)
60. Hochman, M.: Upcrossing inequalities for stationary sequences and applications. The Annals of Probability **37**(6), 2135–2149 (2009)
61. Hölzl, R., Shafer, P.: Universality, optimality, and randomness deficiency. Annals of Pure and Applied Logic **166**(10), 1049–1069 (2015)
62. Hoyrup, M.: Randomness and the ergodic decomposition. In: B. Löwe, D. Normann, I.N. Soskov, A.A. Soskova (eds.) Models of Computation in Context, *Lecture Notes in Computer Science*, vol. 6735, pp. 122–131. Springer, Heidelberg (2011)
63. Hoyrup, M.: The dimension of ergodic random sequences. In: C. Dürr, T. Wilke (eds.) 29th International Symposium on Theoretical Aspects of Computer Science, *LIPIcs. Leibniz Int. Proc. Inform.*, vol. 14, pp. 567–576. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern (2012)
64. Hoyrup, M.: Computability of the ergodic decomposition. Annals of Pure and Applied Logic **164**(5), 542–549 (2013)
65. Hoyrup, M.: Irreversible computable functions. In: E.W. Mayr, N. Portier (eds.) 31st International Symposium on Theoretical Aspects of Computer Science (STACS 2014), *Leibniz International Proceedings in Informatics (LIPIcs)*, vol. 25, pp. 362–373. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2014)
66. Hoyrup, M., Rojas, C.: An application of Martin-Löf randomness to effective probability theory. In: K. Ambos-Spies, B. Löwe, W. Merkle (eds.) Mathematical Theory and Computational Practice, *Lecture Notes in Computer Science*, vol. 5635, pp. 260–269. Springer, Berlin (2009)
67. Hoyrup, M., Rojas, C.: Applications of effective probability theory to Martin-Löf randomness. In: S. Albers, A. Marchetti-Spaccamela, Y. Matias, S.E. Nikoletseas, W. Thomas (eds.) Automata, Languages and Programming. Part I, *Lecture Notes in Comput. Sci.*, vol. 5555, pp. 549–561. Springer, Berlin (2009)

68. Hoyrup, M., Rojas, C.: Computability of probability measures and Martin-Löf randomness over metric spaces. Information and Computation **207**(7), 830–847 (2009)
69. Hoyrup, M., Rojas, C., Weihrauch, K.: Computability of the Radon-Nikodym derivative. In: B. Löwe, D. Normann, I. Soskov, A. Soskova (eds.) Models of Computation in Context, *Lecture Notes in Computer Science*, vol. 6735, pp. 132–141. Springer, Heidelberg (2011)
70. Hoyrup, M., Rojas, C., Weihrauch, K.: Computability of the Radon-Nikodym derivative. Computability **1**(1), 3–13 (2012)
71. Jafarikhah, T., Weihrauch, K.: The Riesz representation operator on the dual of $C[0;1]$ is computable. Journal of Universal Computer Science **19**(6), 750–770 (2013)
72. Kakutani, S., Petersen, K.: The speed of convergence in the ergodic theorem. Monatshefte für Mathematik **91**(1), 11–18 (1981)
73. Kawamura, A.: Computational complexity in analysis and geometry. Ph.D. thesis. Department of Computer Science, University of Toronto (2011)
74. Kawamura, A., Cook, S.A.: Complexity theory for operators in analysis. ACM Transactions on Computation Theory **4**(2), 5:1–5:24 (2012)
75. Kjos-Hanssen, B.: Infinite subsets of random sets of integers. Mathematical Research Letters **16**(1), 103–110 (2009)
76. Kjos-Hanssen, B.: The probability distribution as a computational resource for randomness testing. Journal of Logic and Analysis **2**, Paper 10, 13 (2010)
77. Kjos-Hanssen, B., Nerode, A.: The law of the iterated logarithm for algorithmically random Brownian motion. In: S.N. Artemov, A. Nerode (eds.) Logical Foundations of Computer Science, *Lecture Notes in Computer Science*, vol. 4514, pp. 310–317. Springer, Berlin (2007), Proceedings of the International Symposium (LFCS 2007) held at the City University of New York, New York, June 4–7, 2007
78. Kjos-Hanssen, B., Nguyen, P.K.L.V., Rute, J.: Algorithmic randomness for Doob's martingale convergence theorem in continuous time. Logical Methods in Computer Science **10**(4:12), 1–35 (2014)
79. Ko, K.I.: Complexity Theory of Real Functions. *Progress in Theoretical Computer Science*. Birkhäuser, Boston (1991)
80. van Lambalgen, M.: Random sequences. Ph.D. thesis. Department of Mathematics, University of Amsterdam (1987)
81. Levin, L.A.: Uniform tests of randomness. Soviet Math. Dokl. **17**(2), 337–340 (1976)
82. Li, M., Vitányi, P.: An Introduction to Kolmogorov Complexity and its Applications, 4th edn. *Texts in Computer Science*. Springer, Cham (2019)
83. Lu, H., Weihrauch, K.: Computable Riesz representation for the dual of $C[0;1]$. Mathematical Logic Quarterly **53**(4–5), 415–430 (2007)
84. Lu, H., Weihrauch, K.: Computable Riesz representation for locally compact Hausdorff spaces. Journal of Universal Computer Science **14**(6), 845–860 (2008)
85. Martin-Löf, P.: The definition of random sequences. Information and Control **9**, 602–619 (1966)
86. Miyabe, K.: The difference between optimality and universality. Logic Journal of IGPL **20**, 222–234 (2011)
87. Miyabe, K.: L^1-computability, layerwise computability and Solovay reducibility. Computability **2**(1), 15–29 (2013)
88. Miyabe, K.: Algorithmic randomness over general spaces. Mathematical Logic Quarterly **60**(3), 184–204 (2014)
89. Miyabe, K., Nies, A., Zhang, J.: Using almost-everywhere theorems from analysis to study randomness. The Bulletin of Symbolic Logic **22**(3), 305–331 (2016)
90. Miyabe, K., Rute, J.: Van Lambalgen's theorem for uniformly relative Schnorr and computable randomness. In: R. Downey, J. Brendle, R. Goldblatt, B. Kim (eds.) Proceedings of the 12th Asian Logic Conference, pp. 251–270. World Scientifc Publishing, Hackensack, NJ (2013), held in Wellington, December 15–20, 2011
91. Moser, P.: On the convergence of Fourier series of computable Lebesgue integrable functions. In: R. Dillhage, T. Grubba, A. Sorbi, K. Weihrauch, N. Zhong (eds.) Proceedings of the

Fourth International Conference on Computability and Complexity in Analysis (CCA 2007), *Electronic Notes in Theoretical Computer Science*, vol. 202, pp. 13–18. Elsevier (2008), CCA 2007, Siena, Italy, June 16–18, 2007
92. Müller, N.T.: Computability on random variables. Theoretical Computer Science **219**(1-2), 287–299 (1999)
93. Nakamura, M.: Ergodic theorems for algorithmically random sequences. In: M.J. Dinneen, U. Speidel, D.P. Taylor (eds.) Proceedings of the IEEE ITSOC Information Theory Workshop 2005 on Coding and Complexity ITW 2005, pp. 147–150 (2005)
94. Nandakumar, S.: An effective ergodic theorem and some applications. In: STOC'08: Proceedings of the 40th Annual ACM Symposium on Theory of Computing, pp. 39–44. ACM, New York (2008)
95. Nies, A.: Computability and Randomness, *Oxford Logic Guides*, vol. 51. Oxford University Press, New York (2009)
96. Nies, A.: Differentiability of polynomial time computable functions. In: E.W. Mayr, N. Portier (eds.) 31st International Symposium on Theoretical Aspects of Computer Science (STACS 2014), *Leibniz International Proceedings in Informatics (LIPIcs)*, vol. 25, pp. 602–613. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2014)
97. Pathak, N., Rojas, C., Simpson, S.G.: Schnorr randomness and the Lebesgue differentiation theorem. Proceedings of the American Mathematical Society **142**(1), 335–349 (2014)
98. Phelps, R.R.: Lectures on Choquet's Theorem, 2nd edn. Springer, Berlin, London (2001), previous ed.: Princeton, N.J. : Van Nostrand, 1966
99. Pour-El, M.B., Richards, J.I.: L^p-computability in recursive analysis. Proceedings of the American Mathematical Society **92**, 93–97 (1984)
100. Pour-El, M.B., Richards, J.I.: Computability in Analysis and Physics. *Perspectives in Mathematical Logic*. Springer, Berlin (1989)
101. Rumyantsev, A., Shen, A.: Probabilistic constructions of computable objects and a computable version of Lovász Local Lemma. Fundamenta Informaticae **132**(1), 1–14 (2014)
102. Rute, J.: Computable randomness and betting for computable probability spaces. Mathematical Logic Quarterly **62**(4-5), 335–366 (2016)
103. Rute, J.: When does randomness come from randomness? Theoretical Computer Science **635**, 35–50 (2016)
104. Rute, J.: Schnorr randomness for noncomputable measures. Information and Computation **258**, 50–78 (2018)
105. Šanin, N.: Constructive Real Numbers and Constructive Function Spaces, *Translations of Mathematical Monographs*, vol. 21. American Mathematical Society, Providence (1968)
106. Schnorr, C.P.: Zufälligkeit und Wahrscheinlichkeit, *Lecture Notes in Mathematics*, vol. 218. Springer, Berlin (1971)
107. Schröder, M.: Admissible representations for probability measures. Mathematical Logic Quarterly **53**(4–5), 431–445 (2007)
108. Spitters, B.: A constructive view on ergodic theorems. The Journal of Symbolic Logic **71**(2), 611–623 (2006)
109. Takahashi, H.: On a definition of random sequences with respect to conditional probability. Information and Computation **206**, 1375–1382 (2008)
110. Takahashi, H.: Algorithmic randomness and monotone complexity on product space. Information and Computation **209**(2), 183–197 (2011)
111. Vovk, V.G.: The law of the iterated logarithm for random Kolmogorov, or chaotic, sequences. Theory of Probability and Applications **32**, 413–425 (1987)
112. V'yugin, V.V.: Effective convergence in probability and an ergodic theorem for individual random sequences. SIAM Theory of Probability and Its Applications **42**(1), 39–50 (1997)
113. V'yugin, V.V.: Ergodic theorems for individual random sequences. Theoretical Computer Science **207**(4), 343–361 (1998)
114. V'yugin, V.V.: Non-robustness property of the individual ergodic theorem. Problems of Information Transmission **37**(2), 27–39 (2001)
115. Weihrauch, K.: Computability on the probability measures on the Borel sets of the unit interval. Theoretical Computer Science **219**, 421–437 (1999)

116. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)
117. Weihrauch, K.: On computable metric spaces Tietze-Urysohn extension is computable. In: J. Blanck, V. Brattka, P. Hertling (eds.) Computability and Complexity in Analysis, *Lecture Notes in Computer Science*, vol. 2064, pp. 357–368. Springer, Berlin (2001), 4th International Workshop, CCA 2000, Swansea, UK, September 2000
118. Weihrauch, K.: Computability on measurable functions. Computability **6**(1), 79–104 (2017)
119. Weihrauch, K., Tavana-Roshandel, N.: Representations of measurable sets in computable measure theory. Logical Methods in Computer Science **10**, 3:7,21 (2014)
120. Wu, Y., Ding, D.: Computability of measurable sets via effective topologies. Archive for Mathematical Logic **45**(3), 365–379 (2006)
121. Wu, Y., Weihrauch, K.: A computable version of the Daniell-Stone theorem on integration and linear functionals. Theoretical Computer Science **359**(1–3), 28–42 (2006)
122. Zhong, N., Zhang, B.Y.: L^p-computability. Mathematical Logic Quarterly **45**(4), 449–456 (1999)

Chapter 8
Algorithmic Fractal Dimensions in Geometric Measure Theory

Jack H. Lutz and Elvira Mayordomo

Abstract The development of algorithmic fractal dimensions in this century has had many fruitful interactions with geometric measure theory, especially fractal geometry in Euclidean spaces. We survey these developments, with emphasis on connections with computable functions on the reals, recent uses of algorithmic dimensions in proving new theorems in classical (non-algorithmic) fractal geometry, and directions for future research.

8.1 Introduction

In early 2000, classical Hausdorff dimension [32] was shown to admit a new characterization in terms of betting strategies called martingales [51]. This characterization enabled the development of various effective, i.e., algorithmic, versions of Hausdorff dimension obtained by imposing computability and complexity constraints on these martingales. These algorithmic versions included resource-bounded dimensions, which impose dimension structure on various complexity classes [52], the (constructive) dimensions of infinite binary sequences, which interact usefully with algorithmic information theory [53], and the finite-state dimensions of infinite binary sequences, which interact usefully with data compression and Borel normality [19]. Soon thereafter, classical packing dimension [96, 94] was shown to admit a new characterization in terms of martingales that is exactly dual to the martingale characterization of Hausdorff dimension [1]. This led immediately to the develop-

Jack H. Lutz
Department of Computer Science, Iowa State University, Ames, Iowa, USA
e-mail: lutz@cs.iastate.edu

Elvira Mayordomo
Departamento de Informática e Ingeniería de Sistemas, Instituto de Investigación en Ingeniería de Aragón, Universidad de Zaragoza, Spain
e-mail: elvira@unizar.es

V. Brattka, P. Hertling (eds.), *Handbook of Computability and Complexity in Analysis*,
Theory and Applications of Computability, https://doi.org/10.1007/978-3-030-59234-9_8

ment of strong resource-bounded dimensions, strong (constructive) dimension, and strong finite-state dimension [1], which are all algorithmic versions of packing dimension. In the years since these developments, hundreds of research papers by many authors have deepened our understanding of these algorithmic dimensions.

Most work to date on effective dimensions has been carried out in the Cantor space, which consists of all infinite binary sequences. This is natural, because effective dimensions speak to many issues that were already being investigated in the Cantor space. However, the classical fractal dimensions from which these effective dimensions arose – Hausdorff dimension and packing dimension – are powerful quantitative tools of geometric measure theory that have been most useful in Euclidean spaces and other metric spaces that have far richer structures than the totally disconnected Cantor space.

This chapter surveys research results to date on algorithmic fractal dimensions in geometric measure theory, especially fractal geometry in Euclidean spaces. This is a small fraction of the existing body of work on algorithmic fractal dimensions, but it is substantial, and it includes some exciting new results.

It is natural to identify a real number with its binary expansion and to use this identification to define algorithmic dimensions in Euclidean spaces in terms of their counterparts in the Cantor space. This approach works for some purposes, but it becomes a dead end when algorithmic dimensions are used in geometric measure theory and computable analysis. The difficulty, first noted by Turing in his famous correction [98], is that many *obviously* computable functions on the reals (e.g., addition) are *not* computable if reals are represented by their binary expansions [100]. We thus take a principled approach from the beginning, developing algorithmic dimensions in Euclidean spaces in terms of the quantity $\mathrm{K}_r(x)$ in the following paragraph, so that the theory can seamlessly advance to sophisticated applications.

Algorithmic dimension and strong algorithmic dimension are the most extensively investigated effective dimensions. One major reason for this is that these algorithmic dimensions were shown by the second author and others [68, 1, 57] to have characterizations in terms of Kolmogorov complexity, the central notion of algorithmic information theory. In Section 8.2 below we give a brief introduction to the Kolmogorov complexity $\mathrm{K}_r(x)$ of a point x in Euclidean space at a given precision r.

In Section 8.3 we use the above Kolmogorov complexity notion to develop the algorithmic dimension $\dim(x)$ and the strong algorithmic dimension $\mathrm{Dim}(x)$ of each point x in Euclidean space. This development supports the useful intuition that these dimensions are asymptotic measures of the density of algorithmic information in the point x. We discuss how these dimensions relate to the local dimensions that arise in the so-called thermodynamic formalism of fractal geometry; we discuss the history and terminology of algorithmic dimensions; we review the prima facie case that algorithmic dimensions are geometrically meaningful; and we discuss what is known about the circumstances in which algorithmic dimensions agree with their classical counterparts. We then discuss the authors' use of algorithmic dimensions to analyze self-similar fractals [57]. This analysis gives us a new, information-theoretic proof of the classical formula of Moran [73] for the Hausdorff dimensions of self-

similar fractals in terms of the contraction ratios of the iterated function systems that generate them. This new proof gives a clear account of "where the dimension comes from" in the construction of such fractals. Section 8.3 concludes with a survey of the dimensions of points on lines in Euclidean spaces, a topic that has been surprisingly challenging until a very recent breakthrough by N. Lutz and Stull [63].

We survey interactive aspects of algorithmic fractal dimensions in Euclidean spaces in Section 8.4, starting with the mutual algorithmic dimensions developed by Case and the first author [13]. These dimensions, $\mathrm{mdim}(x : y)$ and $\mathrm{Mdim}(x : y)$, are analogous to the mutual information measures of Shannon information theory and algorithmic information theory. Intuitively, $\mathrm{mdim}(x : y)$ and $\mathrm{Mdim}(x : y)$ are asymptotic measures of the density of the algorithmic information shared by points x and y in Euclidean spaces. We survey the fundamental properties of these mutual dimensions, which are analogous to those of their information-theoretic analogs. The most important of these properties are those that govern how mutual dimensions are affected by functions on Euclidean spaces that are computable in the sense of computable analysis [100]. Specifically, we review the information processing inequalities of [13], which state that $\mathrm{mdim}(f(x) : y) \leq \mathrm{mdim}(x : y)$ and $\mathrm{Mdim}(f(x) : y) \leq \mathrm{Mdim}(x : y)$ hold for all computable Lipschitz functions f, i.e., that applying such a function f to a point x cannot increase the density of algorithmic information that it contains about a point y. We also survey the conditional dimensions $\dim(x|y)$ and $\mathrm{Dim}(x|y)$ recently developed by the first author and N. Lutz [56]. Roughly speaking, these conditional dimensions quantify the density of algorithmic information in x beyond what is already present in y.

It is rare for the theory of computing to be used to answer open questions in mathematical analysis whose statements do not involve computation or related aspects of logic. In Section 8.5 we survey exciting new developments that do exactly this. We first describe new characterizations by the first author and N. Lutz [56] of the classical Hausdorff and packing dimensions of arbitrary sets in Euclidean spaces in terms of the relativized dimensions of the individual points that belong to them. These characterizations are called point-to-set principles because they enable one to use a bound on the relativized dimension of a single, judiciously chosen point x in a set E in Euclidean space to prove a bound on the classical Hausdorff or packing dimension of the set E. We illustrate the power of the point-to-set principle by giving an overview of its use in the new, information-theoretic proof [56] of Davies's 1971 theorem stating that the Kakeya conjecture holds in the Euclidean plane [20]. We then discuss two very recent uses of the point-to-set principle to solve open problems in classical fractal geometry. These are N. Lutz and D. Stull's strengthened lower bounds on the Hausdorff dimensions of generalized Furstenberg sets [63] and N. Lutz's extension of the fractal intersection formulas for Hausdorff and packing dimensions in Euclidean spaces from Borel sets to arbitrary sets. These are, to the best of our knowledge, the first uses of algorithmic information theory to solve open problems in classical mathematical analysis.

We briefly survey promising directions for future research in Section 8.6. These include extending the algorithmic analysis of self-similar fractals [57] to other classes of fractals, extending algorithmic dimensions to metric spaces other than

Euclidean spaces, investigating algorithmic fractal dimensions that are more effective than constructive dimensions (e.g., polynomial-time or finite-state fractal dimensions) in fractal geometry, and extending algorithmic methods to rectifiability and other aspects of geometric measure theory that do not necessarily concern fractal geometry. In each of these we begin by describing an existing result that sheds light on the promise of further inquiry.

Overviews of algorithmic dimensions in the Cantor space appear in [23, 69], though these are already out of date. Even prior to the development of algorithmic fractal dimensions, a rich network of relationships among gambling strategies, Hausdorff dimension, and Kolmogorov complexity was uncovered by reserach of Ryabko [79, 80, 81, 82], Staiger [89, 90, 91], and Cai and Hartmanis [11]. A brief account of this "prehistory" of algorithmic fractal dimensions appears in section 6 of [53].

8.2 Algorithmic Information in Euclidean Spaces

Algorithmic information theory has most often been used in the set $\{0,1\}^*$ of all finite binary strings. The *conditional Kolmogorov complexity* (or *conditional algorithmic information content*) of a string $x \in \{0,1\}^*$ *given* a string $y \in \{0,1\}^*$ is

$$\mathrm{K}(x|y) = \min\{|\pi| \,|\, \pi \in \{0,1\}^* \text{ and } U(\pi,y) = x\}.$$

Here U is a fixed universal Turing machine and $|\pi|$ is the length of a binary "program" π. Hence $\mathrm{K}(x|y)$ is the minimum number of bits required to specify x to U, when y is provided as side information. We refer the reader to any of the standard texts [49, 23, 75, 87] for the history and intuition behind this notion, including its essential invariance with respect to the choice of the universal Turing machine U. The *Kolmogorov complexity* (or *algorithmic information content*) of a string $x \in \{0,1\}^*$ is then

$$\mathrm{K}(x) = \mathrm{K}(x|\lambda),$$

where λ is the empty string.

Routine binary encoding enables one to extend the definitions of $\mathrm{K}(x)$ and $\mathrm{K}(x|y)$ to situations where x and y range over other countable sets such as $\mathbb{N}$, $\mathbb{Q}$, $\mathbb{N} \times \mathbb{Q}$, etc.

The key to "lifting" algorithmic information theory notions to Euclidean spaces is to define the *Kolmogorov complexity* of a set $E \subseteq \mathbb{R}^n$ to be

$$\mathrm{K}(E) = \min\{\mathrm{K}(q) \,|\, q \in \mathbb{Q}^n \cap E\}. \tag{8.1}$$

(Shen and Vereshchagin [88] used a very similar notion for a very different purpose.) Note that $\mathrm{K}(E)$ is the amount of information required to specify not the set E itself, but rather *some* rational point in E. In particular, this implies that

$$E \subseteq F \implies \mathrm{K}(E) \geq \mathrm{K}(F).$$

Note also that, if E contains no rational point, then $\mathrm{K}(E) = \infty$.

The *Kolmogorov complexity* of a point $x \in \mathbb{R}^n$ *at precision* $r \in \mathbb{N}$ is

$$\mathrm{K}_r(x) = \mathrm{K}(\mathrm{B}_{2^{-r}}(x)), \tag{8.2}$$

where $\mathrm{B}_\varepsilon(x)$ is the open ball of radius ε about x, i.e., the number of bits required to specify *some* rational point $q \in \mathbb{Q}^n$ satisfying $|q-x| < 2^{-r}$, where $|q-x|$ is the Euclidean distance of $q-x$ from the origin.

8.3 Algorithmic Dimensions

8.3.1 Dimensions of Points

We now define the *(constructive) dimension* of a point $x \in \mathbb{R}^n$ to be

$$\dim(x) = \liminf_{r\to\infty} \frac{\mathrm{K}_r(x)}{r} \tag{8.3}$$

and the *strong (constructive) dimension* of x to be

$$\mathrm{Dim}(x) = \limsup_{r\to\infty} \frac{\mathrm{K}_r(x)}{r}. \tag{8.4}$$

We note that $\dim(x)$ and $\mathrm{Dim}(x)$ were originally defined in terms of algorithmic betting strategies called gales [53, 1]. The identities (8.3) and (8.4) were subsequent *theorems* proven in [57], refining very similar results in [68, 1]. These identities have been so convenient for work in Euclidean space that it is now natural to regard them as definitions.

Since $\mathrm{K}_r(x)$ is the amount of information required to specify a rational point that approximates x to within 2^{-r} (i.e., with r bits of precision), $\dim(x)$ and $\mathrm{Dim}(x)$ are intuitively the *lower* and *upper asymptotic densities of information* in the point x. This intuition is a good starting point, but the fact that $\dim(x)$ and $\mathrm{Dim}(x)$ are *geometrically* meaningful will only become evident in light of the mathematical consequences of (8.3) and (8.4) surveyed in this chapter.

It is an easy exercise to show that, for all $x \in \mathbb{R}^n$,

$$0 \le \dim(x) \le \mathrm{Dim}(x) \le n. \tag{8.5}$$

If x is a computable point in $\mathbb{R}^n$, then $\mathrm{K}_r(x) = o(r)$, so $\dim(x) = \mathrm{Dim}(x) = 0$. On the other hand, if x is a random point in $\mathbb{R}^n$ (i.e., a point that is algorithmically random in the sense of Martin-Löf [65]), then $\mathrm{K}_r(x) = nr - O(1)$, so $\dim(x) = \mathrm{Dim}(x) = n$. Hence the dimensions of points range between 0 and the dimension of the Euclidean space that they inhabit. In fact, for *every* real number $\alpha \in [0,n]$, the *dimension level set*

$$\mathrm{DIM}^{\alpha} = \{x \in \mathbb{R}^n \,|\, \dim(x) = \alpha\} \tag{8.6}$$

and the *strong dimension level set*

$$\mathrm{DIM}^{\alpha}_{\mathrm{str}} = \{x \in \mathbb{R}^n \,|\, \mathrm{Dim}(x) = \alpha\} \tag{8.7}$$

are uncountable and dense in $\mathbb{R}^n$ [53, 1]. The dimensions $\dim(x)$ and $\mathrm{Dim}(x)$ can coincide, but they do not generally do so. In fact, the set $\mathrm{DIM}^0 \cap \mathrm{DIM}^n_{\mathrm{str}}$ is a comeager (i.e., topologically large) subset of $\mathbb{R}^n$ [37].

Classical fractal geometry has *local*, or *pointwise*, dimensions that are useful, especially in connection with dynamical systems. Specifically, if ν is an *outer measure* on $\mathbb{R}^n$, i.e., a function $\nu : \mathscr{P}(\mathbb{R}^n) \to [0,\infty]$ satisfying $\nu(\emptyset) = 0$, monotonicity ($E \subseteq F \implies \nu(E) \leq \nu(F)$), and countable subadditivity ($E \subseteq \cup_{k=0}^{\infty} E_k \implies \nu(E) \leq \sum_{k=0}^{\infty} \nu(E_k)$), and if ν is *locally finite* (i.e., every $x \in \mathbb{R}^n$ has a neighborhood N with $\nu(N) < \infty$), then the *lower* and *upper local dimensions* of ν at a point $x \in \mathbb{R}^n$ are

$$(\dim_{\mathrm{loc}} \nu)(x) = \liminf_{r\to\infty} \frac{\log(\frac{1}{\nu(\mathrm{B}_{2^{-r}(x)})})}{r} \tag{8.8}$$

and

$$(\mathrm{Dim}_{\mathrm{loc}} \nu)(x) = \limsup_{r\to\infty} \frac{\log(\frac{1}{\nu(\mathrm{B}_{2^{-r}(x)})})}{r}, \tag{8.9}$$

respectively, where $\log = \log_2$ [25].

Until very recently, no relationship was known between the dimensions $\dim(x)$ and $\mathrm{Dim}(x)$ and the local dimensions (8.8) and (8.9). However, N. Lutz recently observed that a very non-classical choice of the outer measure ν remedies this. For each $E \subseteq \mathbb{R}^n$, let

$$\kappa(E) = 2^{-\mathrm{K}(E)}, \tag{8.10}$$

where $\mathrm{K}(E)$ is defined as in (8.1). Then κ is easily seen to be an outer measure on $\mathbb{R}^n$ that is *finite* (i.e., $\kappa(\mathbb{R}^n) < \infty$), hence certainly locally finite, whence the local dimensions $\dim_{\mathrm{loc}} \kappa$ and $\mathrm{Dim}_{\mathrm{loc}} \kappa$ are well defined. In fact we have the following.

Theorem 8.3.1 (N. Lutz [60]). *For all $x \in \mathbb{R}^n$,*

$$\dim(x) = (\dim_{\mathrm{loc}} \kappa)(x)$$

and

$$\mathrm{Dim}(x) = (\mathrm{Dim}_{\mathrm{loc}} \kappa)(x).$$

There is a direct conceptual path from the classical Hausdorff and packing dimensions to the dimensions of points defined in (8.3) and (8.4).

The *Hausdorff dimension* $\dim_{\mathrm{H}}(E)$ of a set $E \subseteq \mathbb{R}^n$ was introduced by Hausdorff [32] before 1920 and is arguably the most important notion of fractal dimension. Its classical definition, which may be found in standard texts such as [93, 25, 7], involves covering the set E by families of sets with diameters vanishing in the limit. In all cases, $0 \leq \dim_{\mathrm{H}}(E) \leq n$.

At the beginning of the present century, in order to formulate versions of Hausdorff dimensions that would work in complexity classes and other algorithmic settings, the first author [52] gave a new characterization of Hausdorff dimension in terms of betting strategies, called gales, on which it is easy to impose computability and complexity conditions. Of particular interest here, he then defined the *constructive dimension* $\mathrm{cdim}(E)$ of a set $E \subseteq \mathbb{R}^n$ exactly like the gale characterization of $\dim_{\mathrm{H}}(E)$, except that the gales were now required to be lower semicomputable [53]. He then defined the dimension $\dim(x)$ of a point $x \in \mathbb{R}^n$ to be the constructive dimension of its singleton, i.e., $\dim(x) = \mathrm{cdim}(\{x\})$. The existence of a universal Turing machine made it immediately evident that constructive dimension has the *absolute stability* property that

$$\mathrm{cdim}(E) = \sup_{x \in E} \dim(x) \tag{8.11}$$

for all $x \in \mathbb{R}^n$. Accordingly, constructive dimension has since been investigated pointwise. As noted earlier, the second author [68] then proved the characterization (8.3) as a theorem.

Two things should be noted about the preceding paragraph. First, these early papers were written entirely in terms of binary sequences, rather than points in Euclidean space. However, the most straightforward binary encoding of points bridges this gap. (In this survey we freely use those results from the Cantor space that do extend easily to Euclidean space.) Second, although the gale characterization is essential for polynomial time and many other stringent levels of effectivization, constructive dimension can be defined equivalently by effectivizing Hausdorff's original formulation [77].

8.3.2 The Correspondence Principle

In 2001, the first author conjectured that there should be a *correspondence principle* (a term that Bohr had used analogously in quantum mechanics) assuring us that for sufficiently simple sets $E \subseteq \mathbb{R}^n$, the constructive and classical dimensions agree, i.e.,

$$\mathrm{cdim}(E) = \dim_{\mathrm{H}}(E). \tag{8.12}$$

Hitchcock [34] confirmed this conjecture, proving that (8.12) holds for any set $E \subseteq \mathbb{R}^n$ that is a union of sets that are computably closed, i.e., that are Π^0_1 in Kleene's arithmetical hierarchy. (This means that (8.12) holds for all Σ^0_2 sets, and also for sets that are nonuniform unions of Π^0_1 sets.) Hitchcock also noted that this result is the best possible in the arithmetical hierarchy, because there are Π^0_2 sets E (e.g., $E = \{z\}$, where z is a Martin-Löf random point that is Δ^0_2) for which (8.12) fails.

By (8.11) and (8.12) we have

$$\dim_{\mathrm{H}}(E) = \sup_{x \in E} \dim(x), \tag{8.13}$$

which is a very nonclassical, pointwise characterization of the classical Hausdorff dimensions of sets that are unions of Π^0_1 sets. Since most textbook examples of fractal sets are Π^0_1, (8.13) is a strong preliminary indication that the dimensions of points are geometrically meaningful.

The *packing dimension* $\dim_{\mathrm{P}}(E)$ of a set $E \subseteq \mathbb{R}^n$ was introduced in the early 1980s by Tricot [96] and Sullivan [94]. Its original definition is a bit more involved than that of Hausdorff dimension [25, 7] and implies that $\dim_{\mathrm{H}}(E) \leq \dim_{\mathrm{P}}(E) \leq n$ for all $E \subseteq \mathbb{R}^n$.

After the development of constructive versions of Hausdorff dimension outlined above, Athreya, Hitchcock, and the authors [1] undertook an analogous development for packing dimension. The gale characterization of $\dim_{\mathrm{P}}(E)$ turns out to be exactly dual to that of $\dim_{\mathrm{H}}(E)$, with just one limit superior replaced by a limit inferior. The *strong constructive dimension* $\mathrm{cDim}(E)$ of a set $E \subseteq \mathbb{R}^n$ is defined by requiring the gales to be lower semicomputable, and the *strong dimension* of a point $x \in \mathbb{R}^n$ is $\mathrm{Dim}(x) = \mathrm{cDim}(\{x\})$. The *absolute stability* of strong constructive dimension,

$$\mathrm{cDim}(E) = \sup_{x \in E} \mathrm{Dim}(x), \tag{8.14}$$

holds for all $E \subseteq \mathbb{R}^n$, as does the Kolmogorov complexity characterization (8.4). All this was shown in [1], but a correspondence principle for strong constructive dimension was left open. In fact, Conidis [16] subsequently used a clever priority argument to construct a Π^0_1 set $E \subseteq \mathbb{R}^n$ for which $\mathrm{cDim}(E) \neq \dim_{\mathrm{P}}(E)$. It is still not known whether some simple, logical definability criterion for E implies that $\mathrm{cDim}(E) = \dim_{\mathrm{P}}(E)$. Staiger's proof that regular ω-languages E satisfy this identity is an encouraging step in this direction [92].

8.3.3 Self-Similar Fractals

The first application of algorithmic dimensions to fractal geometry was the authors' investigation of the dimensions of points in self-similar fractals [57]. We give a brief exposition of this work here, referring the reader to [57] for the many missing details.

Self-similar fractals are the most widely known and best understood classes of fractals [25]. Cantor's middle-third set, the von Koch curve, the Sierpiński triangle, and the Menger sponge are especially well known examples of self-similar fractals.

Briefly, a self-similar fractal in a Euclidean space $\mathbb{R}^n$ is generated from an initial nonempty closed set $D \subseteq \mathbb{R}^n$ by an *iterated function system (IFS)*, which is a finite list $S = (S_0, S_1, \ldots, S_{k-1})$ of $k \geq 2$ contracting similarities $S_i : D \to D$. Each of these similarities S_i is *coded* by the symbol i in the alphabet $\Sigma = \{0, \ldots, k-1\}$, and each S_i has a *contraction ratio* $c_i \in (0,1)$. The IFS S is required to satisfy Moran's *open set condition* [73], which says that there is a nonempty open set $G \subseteq D$ whose images $S_i(G)$, for $i \in \Sigma$, are disjoint subsets of G.

For example, the Sierpiński triangle is generated from the set $D \subseteq \mathbb{R}^2$ consisting of the triangle with vertices $v_0 = (0,0)$, $v_1 = (1,0)$, and $v_2 = (1/2, \sqrt{3}/2)$, together with this triangle's interior, by the IFS $S = (S_0, S_1, S_2)$, where each $S_i : D \to D$ is defined by

$$S_i(p) = v_i + \frac{1}{2}(p - v_i)$$

for $p \in D$. Note that $\Sigma = \{0,1,2\}$ and $c_0 = c_1 = c_2 = 1/2$ in this example. Note also that the open set condition is satisfied here by letting G be the topological interior of D. Each infinite sequence $T \in \Sigma^\infty$ *codes* a point $S(T) \in D$ that is obtained by applying the similarities coded by the successive symbols in T in a canonical way. (See Figure 8.1.) The Sierpiński triangle is the *attractor* (or *invariant set*) of S and D, which consists of all points $S(T)$ for $T \in \Sigma^\infty$.

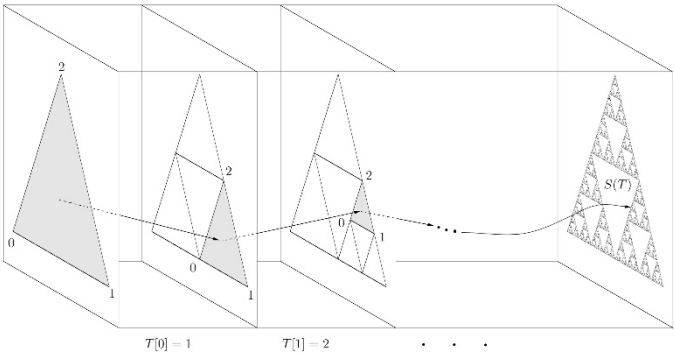


Fig. 8.1 A sequence $T \in \{0,1,2\}^\infty$ codes a point $S(T)$ in the Sierpiński triangle (from [57]).

The main objective of [57] was to relate the dimension and strong dimension of each point $S(T) \in \mathbb{R}^n$ in a self-similar fractal to the corresponding dimensions of the coding sequence T. As it turned out, the algorithmic dimensions in Σ^∞ had to be extended in order to achieve this.

The *similarity dimension* of an IFS $S = (S_0, \ldots, S_{k-1})$ with contraction ratios $c_0, \ldots, c_{k-1} \in (0,1)$ is the unique solution $\mathrm{sdim}(S) = s$ of the equation

$$\sum_{i=0}^{k-1} c_i^s = 1. \tag{8.15}$$

The *similarity probability measure* of S is the probability measure on Σ that is implicit in (8.15), i.e., the function $\pi_S : \Sigma \to [0,1]$ defined by

$$\pi_S(i) = c_i^{\mathrm{sdim}(S)} \tag{8.16}$$

for each $i \in \Sigma$. If the contraction ratios of S are all the same, then π_S is the uniform probability measure on Σ, but this is not generally the case. We extend π_S to the domain Σ^* by setting

$$\pi_S(w) = \prod_{m=0}^{|w|-1} \pi_S(w[m]) \tag{8.17}$$

for each $w \in \Sigma^*$. We define the Shannon *S-self-information* of each string $w \in \Sigma^*$ to be the quantity

$$l_S(w) = \log \frac{1}{\pi_S(w)}. \tag{8.18}$$

Finally, we define the *dimension* of a sequence $T \in \Sigma^\infty$ *with respect to* the IFS S to be

$$\dim^S(T) = \liminf_{j\to\infty} \frac{\mathrm{K}(T[0..j])}{l_S(T[0..j])}. \tag{8.19}$$

Similarly, the *strong dimension* of T *with respect to* S is

$$\mathrm{Dim}^S(T) = \limsup_{j\to\infty} \frac{\mathrm{K}(T[0..j])}{l_S(T[0..j])}. \tag{8.20}$$

The dimension (8.19) is a special case of an algorithmic *Billingsley dimension* [6, 99, 12]. These are treated more generally in [57].

A set $F \subseteq \mathbb{R}^n$ is a *computably self-similar fractal* if it is the attractor of some D and S as above such that the contracting similarities $S_0, \ldots, S_{k-1}$ are all computable in the sense of computable analysis.

The following theorem gives a complete analysis of the dimensions of points in computably self-similar fractals.

Theorem 8.3.2 (J. Lutz and Mayordomo [57]). *If $F \subseteq \mathbb{R}^n$ is a computably self-similar fractal and S is an IFS testifying to this fact, then, for all points $x \in F$ and all coding sequences $T \in \Sigma^\infty$ for x,*

$$\dim(x) = \mathrm{sdim}(S)\dim^S(T) \tag{8.21}$$

and

$$\mathrm{Dim}(x) = \mathrm{sdim}(S)\mathrm{Dim}^S(T). \tag{8.22}$$

The proof of Theorem 8.3.2 is nontrivial. It combines some very strong coding properties of iterated function systems with some geometric Kolmogorov complexity arguments.

The following characterization of continuous functions on the reals is one of the oldest and most beautiful theorems of computable analysis.

Theorem 8.3.3 (Lacombe [45, 46]). *A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is continuous if and only if there is an oracle $A \subseteq \mathbb{N}$ relative to which f is computable.*

Using Lacombe's theorem it is easy to derive the classical analysis of self-similar fractals (which need not be computably self-similar) from Theorem 8.3.2.

Corollary 8.3.4 (Moran [73], Falconer [24]). *For every self-similar fractal $F \subseteq \mathbb{R}^n$ and every IFS S that generates F,*

$$\dim_{\mathrm{H}}(F) = \dim_{\mathrm{P}}(F) = \mathrm{sdim}(F). \tag{8.23}$$

Proof. Let F and S be as given. By Lacombe's theorem there is an oracle $A \subseteq \mathbb{N}$ relative to which S is computable. It follows by a theorem by Kamo and Kawamura [41] that the set F is Π^0_1 relative to A, whence the relativization of (8.13) tells us that

$$\dim^A_{\mathrm{H}}(F) = \sup_{x \in F} \dim^A(x). \tag{8.24}$$

We then have

$$\begin{aligned}
\dim_{\mathrm{H}}(F) &\leq \dim_{\mathrm{P}}(F) \\
&= \dim^A_{\mathrm{P}}(F) \\
&\leq \mathrm{cDim}^A(F) \\
&= \sup_{x \in F} \mathrm{Dim}^A(x) \\
&\overset{(8.22)}{=} \sup_{T \in \Sigma^\infty} \mathrm{sdim}(S)\mathrm{Dim}^{S,A}(T) \\
&= \mathrm{sdim}(S) \\
&= \sup_{T \in \Sigma^\infty} \mathrm{sdim}(S)\dim^{S,A}(T) \\
&\overset{(8.21)}{=} \sup_{x \in F} \dim^A(x) \\
&\overset{(8.24)}{=} \dim^A_{\mathrm{H}}(F) \\
&= \dim_{\mathrm{H}}(F),
\end{aligned}$$

so (8.23) holds. □

Intuitively, Theorem 8.3.2 is stronger than its Corollary 8.3.4, because Theorem 8.3.2 gives a complete account of "where the dimension comes from".

8.3.4 Dimension Level Sets

The dimension level sets DIM^α and $\mathrm{DIM}^\alpha_{\mathrm{str}}$ defined in (8.6) and (8.7) have been the focus of several investigations. It was shown in [53, 1] that, for all $0 \leq \alpha \leq n$,

$$\mathrm{cdim}(\mathrm{DIM}^\alpha) = \dim_{\mathrm{H}}(\mathrm{DIM}^\alpha) = \alpha$$

and

$$\mathrm{cDim}(\mathrm{DIM}^\alpha_{\mathrm{str}}) = \dim_{\mathrm{P}}(\mathrm{DIM}^\alpha_{\mathrm{str}}) = \alpha.$$

Hitchcock, Terwijn, and the first author [33] investigated the complexities of these dimension level sets from the viewpoint of descriptive set theory. Following standard usage [74], we write $\mathbf{\Sigma^0_k}$ and $\mathbf{\Pi^0_k}$ for the classes at the kth level ($k \in \mathbb{Z}^+$) of the Borel hierarchy of subsets of $\mathbb{R}^n$. That is, $\mathbf{\Sigma^0_1}$ is the class of all open subsets of $\mathbb{R}^n$, each $\mathbf{\Pi^0_k}$ is the class of all complements of sets in $\mathbf{\Sigma^0_k}$, and each $\mathbf{\Sigma^0_{k+1}}$ is the class of all countable unions of sets in $\mathbf{\Pi^0_k}$. We also write Σ^0_k and Π^0_k for the classes of the kth level of Kleene's arithmetical hierarchy of subsets of $\mathbb{R}^n$. That is, Σ^0_1 is the class of all computably open subsets of $\mathbb{R}^n$, each Π^0_k is the class of all complements of sets in Σ^0_k, and each Σ^0_{k+1} is the class of all effective (computable) unions of sets in Π^0_k.

Recall that a real number α is *Δ^0_2-computable* if there is a computable function $f : \mathbb{N} \to \mathbb{Q}$ such that $\lim_{k\to\infty} f(k) = \alpha$.

The following facts were proven in [33].

1. DIM^0 is Π^0_2 but not $\mathbf{\Sigma^0_2}$.
2. For all $\alpha \in (0,n]$, DIM^α is $\mathbf{\Pi^0_3}$ (and Π^0_3 if α is Δ^0_2-computable) but not $\mathbf{\Sigma^0_3}$.
3. $\mathrm{DIM}^n_{\mathrm{str}}$ is $\mathbf{\Pi^0_2}$ and Π^0_3 but not $\mathbf{\Sigma^0_2}$.
4. For all $\alpha \in [0,n)$, $\mathrm{DIM}^\alpha_{\mathrm{str}}$ is $\mathbf{\Pi^0_3}$ (and Π^0_4 if α is Δ^0_2-computable) but not $\mathbf{\Sigma^0_3}$.

Weihrauch and the first author [59] investigated the connectivity properties of sets of the form

$$\mathrm{DIM}^I = \bigcup_{\alpha\in I} \mathrm{DIM}^\alpha,$$

where $I \subseteq [0,n]$ is an interval. After making the easy observation that each of the sets $\mathrm{DIM}^{[0,1)}$ and $\mathrm{DIM}^{(n-1,n]}$ is totally disconnected, they proved that each of the sets $\mathrm{DIM}^{[0,1]}$ and $\mathrm{DIM}^{[n-1,n]}$ is path-connected. These results are especially intriguing in the Euclidean plane, where they say that extending either of the sets $\mathrm{DIM}^{[0,1)}$ or $\mathrm{DIM}^{(1,2]}$ to include the level set DIM^1 transforms it from a totally disconnected set to a path-connected set. This suggests that DIM^1 is somehow a very special subset of $\mathbb{R}^2$.

Turetsky [97] investigated this matter further and proved that DIM^1 is a connected set in $\mathbb{R}^n$. He also proved that $\mathrm{DIM}^{[0,1)} \cup \mathrm{DIM}^{(1,2]}$ is not a path-connected subset of $\mathbb{R}^2$.

8.3.5 Dimensions of Points on Lines

Since effective dimension is a pointwise property, it is natural to study the dimension spectrum of a set $E \subseteq \mathbb{R}^n$, i.e., the set $\mathrm{sp}(E) = \{\dim(x) \,|\, x \in E\}$. This study is far from obvious even for sets as apparently simple as straight lines. We review in this section the results obtained so far, mainly for the case of straight lines in $\mathbb{R}^2$.

As noted in Section 8.3.4, the set of points in $\mathbb{R}^2$ of dimension exactly one is connected, while the set of points in $\mathbb{R}^2$ with dimension less than 1 is totally disconnected. Therefore every line in $\mathbb{R}^2$ contains a point of dimension 1. Despite the

surprising fact that there are lines in every direction that contain no random points [55], the first author and N. Lutz have shown that almost every point on any line with random slope has dimension 2 [56]. Still all these results leave open fundamental questions about the structure of the dimension spectra of lines, since they don't even rule out the possibility of a line having the singleton set $\{1\}$ as its dimension spectrum.

Very recently this latest open question has been answered in the negative. N. Lutz and Stull [63] have proven the following general lower bound on the dimension of points on lines in $\mathbb{R}^2$.

Theorem 8.3.5 (N. Lutz and Stull [63]). *For all $a,b,x \in \mathbb{R}$,*

$$\dim(x,ax+b) \geq \dim^{a,b}(x) + \min\{\dim(a,b), \dim^{a,b}(x)\}.$$

In particular, for almost every $x \in \mathbb{R}$, $\dim(x,ax+b) = 1 + \min\{\dim(a,b),1\}$.

Taking $x_1 = 0$ and x_2 a Martin-Löf random real relative to (a,b), Theorem 8.3.5 gives us two points in the line, $(0,b)$ and (x_2, ax_2+b), whose dimensions differ by at least one, so the dimension spectrum cannot be a singleton.

We briefly sketch here the main intuitions behind the (deep) proof of Theorem 8.3.5, fully based on algorithmic information theory. Theorem 8.3.5's aim is to connect $\dim(x,ax+b)$ with $\dim(a,b,x)$ (i.e., a dimension in $\mathbb{R}^2$ with a dimension in $\mathbb{R}^3$). Notice that in the case $\dim(a,b) \leq \dim^{a,b}(x)$ the theorem's conclusion is close to saying $\dim(x,ax+b) \geq \dim(a,b,x)$.

The proof is based on the property that says that under the following two conditions

(i) $\dim(a,b)$ is small
(ii) whenever $ux+v = ax+b$, either $\dim(u,v)$ is large or (u,v) is close to (a,b)

it holds that $\dim(x,ax+b)$ is close to $\dim(a,b,x)$.

There is an extra ingredient to finish this intuition.While condition (ii) can be shown to hold in general, condition (i) can only be proven in a particular relativized world whereas the conclusion of the theorem still holds for every oracle.

N. Lutz and Stull [62] have also shown that the dimension spectrum of a line is always infinite, proving the following two results. The first theorem proves that if $\dim(a,b) = \mathrm{Dim}(a,b)$ then the corresponding line contains a length-1 interval.

Theorem 8.3.6 (N. Lutz and Stull [62]). *Let $a,b \in \mathbb{R}$ satisfy that* $\dim(a,b) = \mathrm{Dim}(a,b)$. *Then for every $s \in [0,1]$ there is a point $x \in \mathbb{R}$ such that* $\dim(x,ax+b) = s + \min\{\dim(a,b),1\}$.

The second result proves that all spectra of lines are infinite.

Theorem 8.3.7 (N. Lutz and Stull [62]). *Let $L_{a,b}$ be any line in $\mathbb{R}^2$. Then the dimension spectrum* $\mathrm{sp}(L_{a,b})$ *is infinite.*

8.4 Mutual and Conditional Dimensions

Just as the dimension of a point x in Euclidean space is the asymptotic density of the algorithmic information in x, the mutual dimension between two points x and y in Euclidean spaces is the asymptotic density of the algorithmic information shared by x and y. In this section, we survey this notion and the data processing inequalities, which estimate the effect of computable functions on mutual dimension. We also survey the related notion of conditional dimension.

8.4.1 Mutual Dimensions

The *mutual (algorithmic) information* between two rational points $p \in \mathbb{Q}^m$ and $q \in \mathbb{Q}^n$ is

$$\mathrm{I}(p:q) = \mathrm{K}(p) - \mathrm{K}(p|q).$$

This notion, essentially due to Kolmogorov [44], is an analog of mutual entropy in Shannon information theory [86, 18, 49]. Intuitively, $\mathrm{K}(p|q)$ is the amount of information in p *not* contained in q, so $\mathrm{I}(p:q)$ is the amount of information in p that *is* contained in q. It is well known [49] that, for all $p \in \mathbb{Q}^m$ and $q \in \mathbb{Q}^n$,

$$\mathrm{I}(p:q) \approx \mathrm{K}(p) + \mathrm{K}(q) - \mathrm{K}(p,q) \tag{8.25}$$

in the sense that the magnitude of the difference between the two sides of (8.25) is $o(\min\{\mathrm{K}(p), \mathrm{K}(q)\})$. This fact is called *symmetry of information*, because it immediately implies that $\mathrm{I}(p:q) \approx \mathrm{I}(q:p)$.

The ideas in the rest of this section were introduced by Case and the first author [13]. In the spirit of (8.1) they defined the *mutual information* between sets $E \subseteq \mathbb{R}^m$ and $F \subseteq \mathbb{R}^n$ to be

$$\mathrm{I}(E:F) = \min\left\{\mathrm{I}(p:q) \,\middle|\, p \in \mathbb{Q}^m \cap E \text{ and } q \in \mathbb{Q}^n \cap F\right\}.$$

This is the amount of information that rational points p and q *must* share in order to be in E and F, respectively. Note that, for all $E_1, E_2 \subseteq \mathbb{R}^m$ and $F_1, F_2 \subseteq \mathbb{R}^n$,

$$\big[(E_1 \subseteq E_2) \text{ and } (F_1 \subseteq F_2)\big] \implies \mathrm{I}(E_1:F_1) \geq \mathrm{I}(E_2:F_2).$$

The *mutual information* between two points $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$ at *precision* $r \in \mathbb{N}$ is

$$\mathrm{I}_r(x:y) = \mathrm{I}(\mathrm{B}_{2^{-r}}(x) : \mathrm{B}_{2^{-r}}(y)).$$

This is the amount of information that rational approximations of x and y *must* share, merely due to their proximities (distance less than 2^{-r}) to x and y.

In analogy with (8.3) and (8.4), the *lower* and *upper mutual dimensions* between points $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^m$ are

$$\mathrm{mdim}(x:y) = \liminf_{r\to\infty} \frac{\mathrm{I}_r(x:y)}{r} \tag{8.26}$$

and

$$\mathrm{Mdim}(x:y) = \limsup_{r\to\infty} \frac{\mathrm{I}_r(x:y)}{r}, \tag{8.27}$$

respectively.

The following theorem shows that the mutual dimensions mdim and Mdim have many of the properties that one should expect them to have. The proof is involved and includes a modest generalization of Levin's coding theorem [47, 48].

Theorem 8.4.1 (Case and J. Lutz [13]). *For all $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$, the following hold.*

1. $\mathrm{mdim}(x:y) \leq \min\{\dim(x), \dim(y)\}$.
2. $\mathrm{Mdim}(x:y) \leq \min\{\mathrm{Dim}(x), \mathrm{Dim}(y)\}$.
3. $\mathrm{mdim}(x:x) = \dim(x)$.
4. $\mathrm{Mdim}(x:x) = \mathrm{Dim}(x)$.
5. $\mathrm{mdim}(x:y) = \mathrm{mdim}(y:x)$.
6. $\mathrm{Mdim}(x:y) = \mathrm{Mdim}(y:x)$.
7. $\dim(x) + \dim(y) - \mathrm{Dim}(x,y) \leq \mathrm{mdim}(x:y) \leq \mathrm{Dim}(x) + \mathrm{Dim}(y) - \mathrm{Dim}(x:y)$.
8. $\dim(x) + \dim(y) - \dim(x,y) \leq \mathrm{Mdim}(x:y) \leq \mathrm{Dim}(x) + \mathrm{Dim}(y) - \dim(x:y)$.
9. *If x and y are independently random, then* $\mathrm{Mdim}(x:y) = 0$.

The expressions $\dim(x,y)$ and $\mathrm{Dim}(x,y)$ in 7 and 8 above refer to the dimensions of the point $(x,y) \in \mathbb{R}^{m+n}$. In 9 above, x and y are *independently random* if (x,y) is a Martin-Löf random point in $\mathbb{R}^{m+n}$.

More properties of mutual dimensions may be found in [13, 14].

8.4.2 Data Processing Inequalities

The data processing inequality of Shannon information theory [18] says that, for any two probability spaces X and Y, any set Z, and any function $f : X \to Z$,

$$\mathrm{I}(f(X);Y) \leq \mathrm{I}(X;Y), \tag{8.28}$$

i.e., the induced probability space $f(X)$ obtained by "processing the information in X through f" has no greater mutual entropy with Y than X has with Y. More succintly, $f(X)$ tells us no more about Y than X tells us about Y. The data processing inequality of algorithmic information theory [49] says that, for any computable partial function $f : \{0,1\}^* \to \{0,1\}^*$, there is a constant $c_f \in \mathbb{N}$ (essentially the number of bits in a program that computes f) such that, for all strings $x \in \mathrm{dom}\, f$ and $y \in \{0,1\}^*$,

$$\mathrm{I}(f(x):y) \leq \mathrm{I}(x:y) + c_f. \tag{8.29}$$

That is, modulo the constant c_f, $f(x)$ contains no more information about y than x contains about y.

The data processing inequality for the mutual dimension mdim should say that every nice function $f : \mathbb{R}^m \to \mathbb{R}^n$ has the property that, for all $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^k$,

$$\mathrm{mdim}(f(x) : y) \leq \mathrm{mdim}(x : y). \tag{8.30}$$

But what should "nice" mean? A nice function certainly should be computable in the sense of computable analysis [10, 43, 100]. But this is not enough. For example, there is a function $f : \mathbb{R} \to \mathbb{R}^2$ that is computable and *space-filling* in the sense that $[0,1]^2 \subseteq \mathrm{range} f$ [83, 17]. For such a function, choose $x \in \mathbb{R}$ such that $\dim(f(x)) = 2$, and let $y = f(x)$. Then

$$\begin{aligned} \mathrm{mdim}(f(x) : y) &= \mathrm{mdim}(y : y) \\ &= \dim(y) \\ &= 2 \\ &> 1 \\ &\geq \dim(x) \\ &\geq \mathrm{mdim}(x : y), \end{aligned}$$

so (8.30) fails.

Intuitively, the above failure of (8.30) occurs because the function f is extremely sensitive to its input, a property that "nice" functions do not have. A function $f : \mathbb{R}^m \to \mathbb{R}^n$ is *Lipschitz* if there is a real number $c > 0$ such that, for all $x_1, x_2 \in \mathbb{R}^m$,

$$|f(x_1) - f(x_2)| \leq c|x_1 - x_2|.$$

The following data processing inequalities show that computable Lipschitz functions are "nice".

Theorem 8.4.2 (Case and J. Lutz [13]). *If $f : \mathbb{R}^m \to \mathbb{R}^n$ is computable and Lipschitz, then, for all $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^k$,*

$$\mathrm{mdim}(f(x) : y) \leq \mathrm{mdim}(x : y)$$

and

$$\mathrm{Mdim}(f(x) : y) \leq \mathrm{Mdim}(x : y).$$

Several more theorems of this type and applications of these appear in [13].

8.4.3 Conditional Dimensions

A comprehensive theory of the fractal dimensions of points in Euclidean spaces requires not only the dimensions $\dim(x)$ and $\mathrm{Dim}(x)$ and the mutual dimensions

$\mathrm{mdim}(x:y)$ and $\mathrm{Mdim}(x:y)$, but also the conditional dimensions $\dim(x|y)$ and $\mathrm{Dim}(x|y)$ formulated by the first author and N. Lutz [56]. We briefly describe these formulations here.

The conditional Kolmogorov complexity $\mathrm{K}(p|q)$, defined for rational points $p \in \mathbb{Q}^m$ and $q \in \mathbb{Q}^n$, is lifted to the conditional dimensions in the following four steps.

1. For $x \in \mathbb{R}^m$, $q \in \mathbb{Q}^n$, and $r \in \mathbb{N}$, the *conditional Kolmogorov complexity of x at precision r given q* is
$$\hat{\mathrm{K}}_r(x|q) = \min\{\mathrm{K}(p|q) \mid p \in \mathbb{Q}^m \cap \mathrm{B}_{2^{-r}}(x)\}.$$
2. For $x \in \mathbb{R}^m$, $y \in \mathbb{R}^n$, and $r,s \in \mathbb{N}$, the *conditional Kolmogorov complexity* of x *at precision r given y at precision s* is
$$\mathrm{K}_{r,s}(x|y) = \max\left\{\hat{\mathrm{K}}_r(x|q) \mid q \in \mathbb{Q}^n \cap \mathrm{B}_{2^{-s}}(y)\right\}.$$
3. For $x \in \mathbb{R}^m$, $y \in \mathbb{R}^n$, and $r \in \mathbb{N}$, the *conditional Kolmogorov complexity* of x *given y* at precision r is
$$\mathrm{K}_r(x|y) = \mathrm{K}_{r,r}(x|y).$$
4. For $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$, the *lower* and *upper conditional dimensions* of x *given y* are
$$\dim(x|y) = \liminf_{r\to\infty} \frac{\mathrm{K}_r(x|y)}{r}$$
and
$$\mathrm{Dim}(x|y) = \limsup_{r\to\infty} \frac{\mathrm{K}_r(x|y)}{r},$$
respectively.

Steps 1, 2, and 4 of the above lifting are very much in the spirit of what has been done in Sections 8.2, 8.3.1, and 8.4.1 above. Step 3 appears to be problematic, because using the same precision bound r for both x and y makes the definition seem arbitrary and "brittle". However, the following result shows that this is not the case.

Theorem 8.4.3 ([56]). *Let $s : \mathbb{N} \to \mathbb{N}$. If $|s(r) - r| = o(r)$, then, for all $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$,*
$$\dim(x|y) = \liminf_{r\to\infty} \frac{\mathrm{K}_{r,s(r)}(x|y)}{r}$$
and
$$\mathrm{Dim}(x|y) = \limsup_{r\to\infty} \frac{\mathrm{K}_{r,s(r)}(x|y)}{r}.$$

The following result is useful for many purposes.

Theorem 8.4.4 (Chain rule for K_r). *For all $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$,*
$$\mathrm{K}_r(x,y) = \mathrm{K}_r(x|y) + \mathrm{K}_r(y) + o(r). \tag{8.31}$$

An *oracle* for a point $y \in \mathbb{R}^n$ is a function $g : \mathbb{N} \to \mathbb{Q}^n$ such that, for all $s \in \mathbb{N}$, $|g(s) - y| \leq 2^{-s}$. The *Kolmogorov complexity* of a rational point $p \in \mathbb{Q}^m$ *relative to* a point $y \in \mathbb{R}^n$ is

$$\mathrm{K}^y(p) = \max\{\mathrm{K}^g(p) \,|\, g \text{ is an oracle for } y\},$$

where $\mathrm{K}^g(p)$ is the Kolmogorov complexity of p when the universal machine has access to the oracle g. The purpose of the maximum here is to prevent $\mathrm{K}^y(p)$ from using oracles g that code more than y into their behaviors. For $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$, the *dimension* $\dim^y(x)$ *relative to* y is defined from $\mathrm{K}^y(p)$ exactly as $\dim(x)$ was defined from $\mathrm{K}(p)$ in Sections 8.2 and 8.3.1 above. The *relativized strong dimension* $\mathrm{Dim}^y(x)$ is defined analogously.

The following result captures the intuition that conditioning on a point y is a restricted form of oracle access to y.

Lemma 8.4.5 ([56]). *For all $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$, $\dim^y(x) \leq \dim(x|y)$ and $\mathrm{Dim}^y(x) \leq \mathrm{Dim}(x|y)$.*

The remaining results in this section confirm that conditional dimensions have the correct information-theoretic relationships to dimensions and mutual dimensions.

Theorem 8.4.6 ([56]). *For all $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$,*

$$\mathrm{mdim}(x : y) \geq \dim(x) - \mathrm{Dim}(x|y)$$

and

$$\mathrm{Mdim}(x : y) \leq \mathrm{Dim}(x) - \dim(x|y).$$

Theorem 8.4.7 (Chain rule for dimension [56]). *For all $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$,*

$$\begin{aligned} \dim(x) + \dim(y|x) &\leq \dim(x,y) \\ &\leq \dim(x) + \mathrm{Dim}(y|x) \\ &\leq \mathrm{Dim}(x,y) \\ &\leq \mathrm{Dim}(x) + \mathrm{Dim}(y|x). \end{aligned}$$

8.5 Algorithmic Discovery of New Classical Theorems

8.5.1 The Point-to-Set Principle

Many of the most challenging problems in geometric measure theory are problems of establishing lower bounds on the classical fractal dimensions $\dim_{\mathrm{H}}(E)$ and $\dim_{\mathrm{P}}(E)$ for sets $E \subseteq \mathbb{R}^n$. Although such problems seem to involve global properties

of the sets E and make no mention of algorithms, the dimensions of points have recently been used to prove new lower bound results for classical fractal dimensions. The key to these developments is the following pair of theorems of the first author and N. Lutz.

Theorem 8.5.1 (Point-to-set-principle for Hausdorff dimension [56]). *For every* $E \subseteq \mathbb{R}^n$,

$$\dim_{\mathrm{H}}(E) = \min_{A \subseteq \mathbb{N}} \sup_{x \in E} \dim^A(x). \tag{8.32}$$

Theorem 8.5.2 (Point-to-set-principle for packing dimension [56]). *For every* $E \subseteq \mathbb{R}^n$,

$$\dim_{\mathrm{P}}(E) = \min_{A \subseteq \mathbb{N}} \sup_{x \in E} \mathrm{Dim}^A(x). \tag{8.33}$$

The relativized dimensions $\dim^A(x)$ and $\mathrm{Dim}^A(x)$ here are defined by substituting $\mathrm{K}^A_r(x)$ for $\mathrm{K}_r(x)$ in (8.3) and (8.4).

It is to be emphasized that these two theorems completely characterize $\dim_{\mathrm{H}}(E)$ and $\dim_{\mathrm{P}}(E)$ for *all* sets $E \subseteq \mathbb{R}^n$. These characterizations are called *point-to-set principles* because they enable one to use a lower bound on the relativized dimension of a single, judiciously chosen point $x \in E$ to establish a lower bound on the classical dimension of the set E itself. More precisely, for example, Theorem 8.5.1 says that, in order to prove a lower bound $\dim_{\mathrm{H}}(E) \geq \alpha$, it suffices to show that, for every oracle $A \subseteq \mathbb{N}$ and every $\varepsilon > 0$, there is a point $x \in E$ such that $\dim^A(x) \geq \alpha - \varepsilon$. In some cases, it can in fact be shown that, for every oracle $A \subseteq \mathbb{N}$, there is a point $x \in E$ such that $\dim^A(x) \geq \alpha$. While the arbitrary oracle A is essential for the correctness of such proofs, the discussion below shows that its presence has not been burdensome in applications to date.

8.5.2 Plane Kakeya Sets

The first application of the point-to-set principle was not a new theorem, but rather a new, information-theoretic proof of an old theorem. We describe this proof here because it illustrates the intuitive power of the point-to-set principle.

A *Kakeya set* in $\mathbb{R}^n$ is a set $K \subseteq \mathbb{R}^n$ that contains a unit segment in every direction. Sometime before 1920, Besicovitch [4, 5] proved the then-surprising existence of Kakeya sets of Lebesgue measure 0 in $\mathbb{R}^n$ for all $n \geq 2$ and asked whether Kakeya sets in $\mathbb{R}^2$ can have dimension less than 2 [20]. The famous *Kakeya conjecture* (in its most commonly stated form) asserts a negative answer to this and the analogous questions in higher dimensions. That is, the Kakeya conjecture says that every Kakeya set in a Euclidean space $\mathbb{R}^n$ has Hausdorff dimension n. This conjecture holds trivially for $n = 1$ and Davies [20] proved that it holds for $n = 2$. The Kakeya conjecture remains an important open problem for $n \geq 3$ [101, 95].

Our objective here is to sketch the new proof by the first author and N. Lutz [56] of Davies's theorem, that the Kakeya conjecture holds in the Euclidean plane $\mathbb{R}^2$.

This proof uses the following lower bound on the dimensions of points in a line $y = mx + b$.

Lemma 8.5.3 (J. Lutz and N. Lutz [56]). *Let $m \in [0,1]$ and $b \in \mathbb{R}$. For almost every $x \in [0,1]$,*

$$\dim(x, mx+b) \geq \liminf_{r\to\infty} \frac{\mathrm{K}_r(m,b,x) - \mathrm{K}_r(b|m)}{r}. \tag{8.34}$$

We do not prove this lemma here, but note that the proof relativizes, so the lemma holds relative to every oracle $A \subseteq \mathbb{N}$.

To prove Davies's theorem, let $\mathrm{K} \subseteq \mathbb{R}^2$ be a Kakeya set. By the point-to-set principle, fix $A \subseteq \mathbb{N}$ such that

$$\dim_{\mathrm{H}}(K) = \sup_{(x,y)\in K} \dim^A(x,y). \tag{8.35}$$

Fix $m \in [0,1]$ such that

$$\dim^A(m) = 1. \tag{8.36}$$

(This holds for any m that is random relative to A.) Since K is Kakeya, there is a unit segment $L \subseteq K$ of slope m. Let (x_0, y_0) be the left endpoint of L, let $q \in \mathbb{Q} \cap [x_0, x_0 + 1/2]$, and let L' be the unit segment of slope m whose endpoint is $(x_0 - q, y_0)$. Then L' crosses the y-axis at the point $b = mq + y_0$. By Lemma 8.5.3 (relativized to A), fix $x \in [0, 1/2]$ such that

$$\dim^{A,m,b}(x) = 1 \tag{8.37}$$

and

$$\dim^A(x, mx+b) \geq \liminf_{r\to\infty} \frac{\mathrm{K}^A_r(m,b,x) - \mathrm{K}^A_r(b|m)}{r}. \tag{8.38}$$

(Such an x exists, because almost every $x \in [0, 1/2]$ satisfies (8.37) and (8.38).)

In the language of Section 8.5.1, our "judiciously chosen point" is $(x+q, mx+b) \in L \subseteq K$, and the point-to-set principle tells us that it suffices to prove that

$$\dim^A(x+q, mx+b) = 2. \tag{8.39}$$

But this is now easy. Since q is rational, (8.38) and two applications of the chain rule (8.31) tell us that

$$\begin{aligned}
\dim^A(x+q,mx+b) &= \dim^A(x,mx+b)\\
&\geq \liminf_{r\to\infty} \frac{\mathrm{K}^A_r(m,b,x)-\mathrm{K}^A_r(b,m)+\mathrm{K}^A_r(m)}{r}\\
&= \liminf_{r\to\infty} \frac{\mathrm{K}^A_r(x|b,m)+\mathrm{K}^A_r(m)}{r}\\
&\geq \liminf_{r\to\infty} \frac{\mathrm{K}^{A,m,b}_r(x)}{r} + \liminf_{r\to\infty} \frac{\mathrm{K}^A_r(m)}{r}\\
&= \dim^{A,m,b}(x)+\dim^A(m),
\end{aligned}$$

whence (8.36) and (8.37) tell us that (8.39) holds.

This information-theoretic proof of Davies can be summarized in very intuitive terms: because K is Kakeya, it contains a unit segment L whose slope m has dimension 1 relative to A. A rational shift of L to a unit segment L' crosses the y-axis at some point b. Lemma 8.5.3 then gives us a point $(x,mx+b)$ on L' that has dimension 2 relative to A. The point on L from which $(x,mx+b)$ was shifted lies in K and also has dimension 2 relative to A, so K has Hausdorff dimension 2.

The following two sections discuss recent uses of this method to prove *new* theorems in classical fractal geometry.

8.5.3 Intersections and Products of Fractals

We now consider two fundamental, nontrivial, textbook theorems of fractal geometry. The first, over thirty years old and called the *intersection formula*, concerns the intersection of one fractal with a random translation of another fractal.

Theorem 8.5.4 (Kahane [40], Mattila [66, 67]). *For all Borel sets $E,F\subseteq\mathbb{R}^n$ and almost every $z\in\mathbb{R}^n$,*

$$\dim_{\mathrm{H}}(E\cap(F+z))\leq\max\{0,\dim_{\mathrm{H}}(E\times F)-n\}.$$

The second theorem, over sixty years old and called the *product formula*, concerns the product of two fractals.

Theorem 8.5.5 (Marstrand [64]). *For all $E\subseteq\mathbb{R}^n$ and $F\subseteq\mathbb{R}^n$,*

$$\dim_{\mathrm{H}}(E\times F)\geq\dim_{\mathrm{H}}(E)+\dim_{\mathrm{H}}(F).$$

In a recent breakthrough, algorithmic dimension was used to prove the following extension of the intersection formula from Borel sets to *all* sets. We include the simple (given the machinery that we have developed) and instructive proof here.

Theorem 8.5.6 (N. Lutz [61]). *For all sets $E,F\subseteq\mathbb{R}^n$ and almost every $z\in\mathbb{R}^n$,*

$$\dim_{\mathrm{H}}(E\cap(F+z))\leq\max\{0,\dim_{\mathrm{H}}(E\times F)-n\}. \tag{8.40}$$

Proof. Let $E, F \subseteq \mathbb{R}^n$ and $z \in \mathbb{R}^n$. The theorem is trivially affirmed if $F+z$ is disjoint from E, so assume not. By the point-to-set principle, fix an oracle $A \subseteq \mathbb{N}$ such that

$$\dim_{\mathrm{H}}(E \times F) = \sup_{(x,y) \in E \times F} \dim^A(x,y). \tag{8.41}$$

Let $\varepsilon > 0$. Since $E \cap (F+z) \neq \emptyset$, the point-to-set principle tells us that there is a point $x \in E \cap (F+z)$ satisfying

$$\dim^{A,z}(x) > \dim_{\mathrm{H}}(E \cap (F+z)) - \varepsilon. \tag{8.42}$$

Now $(x, x-z) \in E \times F$, so (8.41), Theorem 8.3.5, Lemma 8.4.5, and (8.42) tell us that

$$\begin{aligned}
\dim_{\mathrm{H}}(E \times F) &\geq \dim^A(x, x-z) \\
&= \dim^A(x,z) \\
&\geq \dim^A(z) + \dim^A(x|z) \\
&\geq \dim^A(z) + \dim^{A,z}(x) \\
&> \dim^A(z) + \dim_{\mathrm{H}}(E \cap (F+z)) - \varepsilon.
\end{aligned}$$

Since ε is arbitrary here, it follows that

$$\dim_{\mathrm{H}}(E \cap (F+z)) \leq \dim_{\mathrm{H}}(E \times F) - \dim^A(z).$$

Since almost every $z \in \mathbb{R}^n$ is Martin-Löf random relative to A and hence satisfies $\dim^A(z) = n$, this affirms the theorem. □

The paper [61] shows that the same method gives a new proof of the analog of Theorem 8.5.6 for packing dimension. This result was already known to hold for all sets E and F [26], but the new proof makes clear what a strong duality between Hausdorff and packing dimensions is at play in the intersection formulas.

The paper [61] also gives a new, algorithmic proof of the following known extension of Theorem 8.5.5.

Theorem 8.5.7 (Marstrand [64], Tricot [96]). *For all $E \subseteq \mathbb{R}^m$ and $F \subseteq \mathbb{R}^n$,*

$$\begin{aligned}
\dim_{\mathrm{H}}(E) + \dim_{\mathrm{H}}(F) &\leq \dim_{\mathrm{H}}(E \times F) \\
&\leq \dim_{\mathrm{H}}(E) + \dim_{\mathrm{P}}(F) \\
&\leq \dim_{\mathrm{P}}(E \times F) \\
&\leq \dim_{\mathrm{P}}(E) + \dim_{\mathrm{P}}(F).
\end{aligned}$$

This new proof is much simpler than previously known proofs of Theorem 8.5.7, roughly as simple as previously known proofs of the restriction of Theorem 8.5.7 to Borel sets. The new proof is also quite natural, using the point-to-set principle to derive Theorem 8.5.7 from the formally similar Theorem 8.4.7.

8.5.4 Generalized Furstenberg Sets

For $\alpha \in (0,1]$, a plane set $E \subseteq \mathbb{R}^2$ is said to be *of Furstenberg type with parameter* α or, more simply, *α-Furstenberg*, if, for every direction $e \in S^1$ (where S^1 is the unit circle in $\mathbb{R}^2$), there is a line $\mathscr{L}_e$ in direction e such that $\dim_{\mathrm{H}}(\mathscr{L}_e \cap E) \geq \alpha$.

According to Wolff [101], the following well-known bound is probably due to Furstenberg and Katznelson.

Theorem 8.5.8. *For every $\alpha \in (0,1]$, every α-Furstenberg set $E \subseteq \mathbb{R}^2$ satisfies*

$$\dim_{\mathrm{H}}(E) \geq \alpha + \max\{1/2, \alpha\}.$$

Note that every Kakeya set in the plane is 1-Furstenberg (since it contains a line segment, which has Hausdorff dimension 1, in every direction $e \in S^1$), so Davies's theorem follows from the case $\alpha = 1$ of Theorem 8.5.8. It is an open question – one with connections to Falconer's distance conjecture [42] and Kakeya sets [101] – whether Theorem 8.5.8 can be improved.

In 2012, Molter and Rela generalized α-Furstenberg sets in a natural way. For $\alpha, \beta \in (0,1]$, a set $E \subseteq \mathbb{R}^2$ is *(α,β)-generalized Furstenberg* if there is a set $J \subseteq S^1$ such that $\dim_{\mathrm{H}}(J) \geq \beta$ and, for every $e \in J$, there is a line $\mathscr{L}_e$ in direction e such that $\dim_{\mathrm{H}}(\mathscr{L}_e \cap E) \geq \alpha$. They then proved the following lower bound.

Theorem 8.5.9 (Molter and Rela [72]). *For $\alpha, \beta \in (0,1]$, every (α,β)-generalized Furstenberg set $E \subseteq \mathbb{R}^2$ satisfies*

$$\dim_{\mathrm{H}}(E) \geq \max\{\beta/2, \alpha + \beta - 1\}.$$

Note that every α-Furstenberg set is $(\alpha,1)$-generalized Furstenberg, so Theorem 8.5.8 follows from the case $\beta = 1$ of Theorem 8.5.9.

Algorithmic dimensions were recently used to prove the following result, which improves Theorem 8.5.9 when $\alpha, \beta \in (0,1)$ and $\beta < 2\alpha$.

Theorem 8.5.10 (N. Lutz and Stull [63]). *For all $\alpha, \beta \in (0,1]$, every (α,β)-generalized Furstenberg set $E \subseteq \mathbb{R}^2$ satisfies*

$$\dim_{\mathrm{H}}(E) \geq \alpha + \min\{\beta, \alpha\}.$$

The proof of Theorem 8.5.10 uses the point-to-set principle and Theorem 8.3.5.

8.6 Research Directions

8.6.1 Beyond Self-Similarity

In previous sections we have analyzed the dimension of points in self-similar fractals, but interesting natural examples need more elaborate concepts that combine

self-similarity with random selection. In [31] Gu, Moser, and the authors started the more challenging task of analyzing the dimensions of points in random fractals. They focused on fractals that are randomly selected subfractals of a given self-similar fractal.

Let $F \subseteq \mathbb{R}^n$ be a computably self-similar fractal as defined in Section 8.3.3, with $S = (S_0, \ldots, S_{k-1})$ the corresponding IFS, and $\Sigma = \{0, \ldots, k-1\}$. Recall that each point $x \in F$ has a coding sequence $T \in \Sigma^\infty$, meaning that the point x is obtained by applying the similarities coded by the successive symbols in T. We are interested in certain randomly selected subfractals of the fractal F.

The specification of a point in such a subfractal can be formulated as the outcome of an infinite two-player game between a *selector* that selects the subfractal and a *coder* that selects a point within the subfractal. Specifically, the selector selects r out of the k similarities and this choice depends on the coder's earlier choices, that is, a *selector* is a function $\sigma : \Gamma^* \to [\Sigma]^r$ where $[\Sigma]^r$ is the set of all r-element subsets of Σ, alphabet $\Gamma = \{0, \ldots, r-1\}$, and each element in Γ^* represents a coder's earlier history. A *coder* is a sequence $U \in \Gamma^\infty$, that is, the coder selects a point in the subfractal by repeatedly choosing a similarity out of the r previously picked by the selector. Once a selector σ and a coder U have been chosen, the outcome of the selector-coder game is a point determined by the sequence $\sigma * U \in \Sigma^\infty$ that can be precisely defined as

$$(\sigma * U)[t] = \text{“the } U[t]\text{th element of } \sigma(U[0..t-1])\text{”}$$

for all $t \in \mathbb{N}$.

Each selector σ specifies (selects) the subfractal F_σ of F consisting of all points with coding sequence T for which T is an outcome of playing σ against some coder, $F_\sigma = \{S(\sigma * U) \mid U \in \Gamma^\infty\}$.

The focus of [31] is on randomly selected subfractals of F, by which we mean subfractals F_σ of F for which the selector σ is random with respect to some probability measure. That is, we are interested in the case where the coder is playing a “game against nature” (in order to make precise the idea of algorithmically random selector each selector $\sigma : \Gamma^* \to [\Sigma]^r$ is identified with its *characteristic* sequence $\chi_\sigma \in ([\Sigma]^r)^\infty$).

Gu et al. determine the dimension spectra of a wide class of such randomly selected subfractals, showing that each such fractal has a dimension spectrum that is a closed interval whose endpoints can be computed or approximated from the parameters of the fractal. In general, the maximum of the spectrum is determined by the degree to which the coder can *reinforce* the randomness in the selector, while the minimum is determined by the degree to which the coder can *cancel* randomness in the selector. This randomness cancellation phenomenon has also arisen in other contexts, notably dimension spectra of random closed sets [2, 21] and of random translations of the Cantor set [22]. The main result in [31] concerns subfractals that are similarity random, that is, F_σ defined by a selector σ that is $\hat{\pi}_S$-random. Here $\hat{\pi}_S$ is the natural extension of π_s, the similarity probability measure on Σ defined in Section 8.3.3.

Theorem 8.6.1 ([31]). *For every similarity random subfractal F_σ of F, the dimension spectrum* $\mathrm{sp}(F_\sigma)$ *is an interval satisfying*

$$\left[s^* \frac{\log(k-1) - \log(r-1+A(k-r))}{\log \frac{1}{a}}, s^* \right] \subseteq \mathrm{sp}(F_\sigma) \subseteq \left[s^* \frac{\log k - \log r}{\log \frac{1}{A}}, s^* \right],$$

where $s^* = \mathrm{sdim}(S)$, $a = \min\{\pi_S(i) \mid i \in \Sigma\}$, *and* $A = \max\{\pi_S(i) \mid i \in \Sigma\}$.

In particular, if all the contraction ratios of F have the same value c, then every similarity-random (i.e., uniformly random) subfractal F_σ of F has dimension spectrum

$$\mathrm{sp}(F_\sigma) = \left[s^*(1 - \frac{\log r}{\log k}), s^* \right],$$

where $s^* = \mathrm{sdim}(S) = (\log k)/(\log \frac{1}{c})$.

Many challenging open questions remain concerning the analysis of the dimension of points in more general versions of random fractals, both by extending the results in [31] to random selectors for different probability measures and by considering generalizations such as self-affine fractals and fractals with randomly chosen contraction ratios.

8.6.2 Beyond Euclidean Spaces

While Euclidean space has a very well-behaved metric based on a Borel measure μ, where for instance s-Hausdorff measure coincides with μ for $s = 1$, this is not the case for other metric spaces. Since both Hausdorff and packing dimension can be defined in any metric space, the second author has considered in [70] the extension of algorithmic dimension to a large class of separable metric spaces, the class of spaces with a computable nice cover. This extension includes an algorithmic information characterization of constructive dimension, based on the concept of Kolmogorov complexity of a point at a certain precision, which is an extension of the concept presented in Section 8.2 for Euclidean space.

8.6.3 Beyond Computability

Resource-bounded dimension, introduced in [52] by the first author, has been a very fruitful tool in the quantitative study of complexity classes; see [35, 54] for the main results. Many of the main complexity classes have a suitable resource bound for which the corresponding dimension is adequate for the class, since it has maximal value for the whole class.

The development of resource-bounded dimension was based on a characterization of Hausdorff dimension in terms of betting strategies, imposing different

complexity constraints on those strategies to obtain the different resource-bounded dimensions. Contrary to the case of computability constraints introduced in Section 8.3, many important resource bounds such as polynomial-time dimension do not have corresponding algorithmic information characterizations (although more elaborate compression algorithm characterizations have been obtained in [50, 38]).

In fact the study of gambling under very low complexity constraints, finite-state computability, has been studied at least since the 1970s [85, 27] and the corresponding effective dimension, finite-state dimension, was studied by Dai, Lathrop, and the two authors [19] where finite-state dimension is characterized in terms of finite-state compression.

For the definition of resource-bounded dimension, a class of languages $\mathscr{C}$ is represented via characteristic sequences as a set of infinite binary sequences $\mathscr{C} \subseteq \{0,1\}^\infty$. Using binary representation each language can be seen as a real number in $[0,1]$ and resource-bounded dimension as a tool in Euclidean space. Resource-bounded dimension has a natural extension Σ^∞ for other finite alphabets Σ and the first question is therefore whether the choice of alphabet is relevant for the study of Euclidean space. A satisfactory answer is given in [36] where it is proven that polynomial-time dimension is invariant under base change, that is, for every base b and set $X \subseteq \mathbb{R}$ the set of base-b representations of all elements in X has a polynomial-time dimension independent of b.

Finite-state dimension is not closed under base change, but its connections with number theory are deep. Borel introduced normal numbers in [8], defining a real number α to be Borel normal in base b if for every finite sequence w of base-b digits, the asymptotic, empirical frequency of w in the base-b expansion of α is $b^{-|w|}$. There is a tight relationship between Borel normality and finite-state dimension, since a real number is normal in base b iff its base-b representation is a finite-state dimension-1 sequence [85, 9]. It is known [15, 84] that there are numbers that are normal in one base but not in another, so the nonclosure under base change property of finite-state dimension is a corollary of these results. Absolutely normal numbers are real numbers that are normal in every base, so they correspond to real numbers whose base-b representation has finite-state dimension 1 for every base b; this characterization has been used in very effective constructions of absolutely normal numbers [3, 58]. It is natural to ask whether there are real numbers for which the finite-state dimension of their base-b representations is strictly between 0 and 1 and does not depend on the base b.

8.6.4 Beyond Fractals

This chapter's primary focus is the role of algorithmic fractal dimensions in fractal geometry. However, it should be noted that fractal geometry is only a part of geometric measure theory, and that algorithmic methods may shed light on many other aspects of geometric measure theory.

Many questions in geometric measure theory involve rectifiability [28]. The simplest case of this classical notion is the rectifiability of curves. A *curve* in $\mathbb{R}^n$ is a continuous function $f : [0,1] \to \mathbb{R}^n$. The *length* of a curve f is

$$\text{length}(f) = \sup_{\mathbf{a}} \sum_{i=0}^{k-1} |f(a_{i+1}) - f(a_i)|,$$

where the supremum is taken over all *dissections* $\mathbf{a}$ of $[0,1]$, i.e., all $\mathbf{a} = (a_0, \ldots, a_k)$ with $0 = a_0 < a_1 < \ldots < a_k = 1$. Note that $\text{length}(f)$ is the length of the actual path traced by f, which may "retrace" parts of its range. (In fact, there are computable curves f for which every computable curve g with the same range *must* do unboundedly many such retracings [30].) A curve f is *rectifiable* if $\text{length}(f) < \infty$.

Gu and the authors [29] posed the fanciful question, "Where can an infinitely small nanobot go?" Intuitively, the nanobot is the size of a Euclidean point, and its motion is algorithmic, so its trajectory must be a curve $f : [0,1] \to \mathbb{R}^n$ that is computable in the sense of computable analysis [100]. Moreover, the nanobot's trajectory f should be rectifiable. This last assumption, aside from being intuitively reasonable, prevents the question from being trivialized by space-filling curves [83, 17].

The above considerations translate our fanciful question about a nanobot to the following mathematical question. *Which points in $\mathbb{R}^n$ ($n \geq 2$) lie on rectifiable computable curves?* In honor of an anonymous, poetic reviewer who called the set of all such points "the beaten path", we write $\text{BP}^{(n)}$ for the set of all points in $\mathbb{R}^n$ that lie on rectifiable computable curves. The objective of [29] was to characterize the elements of $\text{BP}^{(n)}$.

A few preliminary observations on the set $\text{BP}^{(n)}$ are in order here. Every computable point in $\mathbb{R}^n$ clearly lies in $\text{BP}^{(n)}$, so $\text{BP}^{(n)}$ is a dense subset of $\mathbb{R}^n$. It is also easy to see that $\text{BP}^{(n)}$ is path-connected. On the other hand, the ranges of rectifiable curves have Hausdorff dimension 1 [25] and there are only countably many computable curves, so $\text{BP}^{(n)}$ is a countable union of sets of Hausdorff dimension 1 and hence has Hausdorff dimension 1. Since $n \geq 2$, this implies that most points in $\mathbb{R}^n$ do not lie on the beaten path $\text{BP}^{(n)}$.

For each rectifiable computable curve f, the set range f is a computably closed, i.e., Π^0_1, subset of $\mathbb{R}^n$. By the preceding paragraph and Hitchcock's correspondence principle (8.13), it follows that $\text{cdim}(\text{BP}^{(n)}) = 1$, whence every point $x \in \text{BP}^{(n)}$ satisfies $\dim(x) \leq 1$. This is a necessary, but not sufficient condition for membership in $\text{BP}^{(n)}$, because the complement of $\text{BP}^{(n)}$ contains points of arbitrarily low dimension [29]. Characterizing membership in $\text{BP}^{(n)}$ thus requires algorithmic methods to be extended beyond fractal dimensions.

The "analyst's traveling salesman theorem" of geometric measure theory characterizes those subsets of Euclidean space that are contained in rectifiable curves. This celebrated theorem was proven for the plane by Jones [39] and extended to high-dimensional Euclidean spaces by Okikiolu [76]. The main contribution of [29] is to formulate the notion of a *computable Jones constriction*, an algorithmic version of the infinitary data structure implicit in the analyst's traveling salesman theorem, and to prove the *computable analyst's traveling salesman theorem*, which says that a

point in Euclidean space lies on the beaten path $\mathrm{BP}^{(n)}$ if and only if it is "permitted" by some computable Jones constriction.

The computable analysis of points in rectifiable curves has continued in at least two different directions. In one direction, Rettinger and Zheng have shown (answering a question in [29]) that there are points in $\mathrm{BP}^{(n)}$ that do not lie on any computable curve of computable length [78] and extended this to obtain a four-level hierarchy of simple computable planar curves that are *point-separable* in the sense that the sets of points lying on curves of the four types are distinct [102]. In another direction, McNicholl [71] proved that there is a point on a computable arc (a set computably homeomorphic to $[0, 1]$) that does not lie in $\mathrm{BP}^{(n)}$. In the same paper, McNicholl used a beautiful geometric priority argument to prove that there is a point on a computable curve of computable length that does not lie on any computable arc.

It is apparent from the above results that algorithmic methods will have a great deal more to say about rectifiability and other aspects of geometric measure theory.

Acknowledgements The first author's work was supported in part by National Science Foundation research grants 1247051 and 1545028 and is based in part on lectures that he gave at the New Zealand Mathematical Research Institute Summer School on Mathematical Logic and Computability, January 9-14, 2017. He thanks Neil Lutz for useful discussions and Don Stull and Andrei Migunov for helpful comments on the exposition. The second author's research was supported in part by Spanish Government MEC Grants TIN2011-27479-C04-01 and TIN2016-80347-R and was done and based in part on lectures given during a research stay at the Institute for Mathematical Sciences at the National University of Singapore for the Program on Aspects of Computation, August 2017. We thank two anonymous reviewers for useful suggestions on the exposition.

References

1. Athreya, K.B., Hitchcock, J.M., Lutz, J.H., Mayordomo, E.: Effective strong dimension in algorithmic information and computational complexity. SIAM Journal on Computing **37**(3), 671–705 (2007)
2. Barmpalias, G., Brodhead, P., Cenzer, D., Dashti, S., Weber, R.: Algorithmic randomness of closed sets. J. Log. Comput. **17**(6), 1041–1062 (2007)
3. Becher, V., Heiber, P.A., Slaman, T.A.: A polynomial-time algorithm for computing absolutely normal numbers. Information and Computation **232**, 1–9 (2013)
4. Besicovitch, A.S.: Sur deux questions d'intégrabilité des fonctions. Journal de la Société de physique et de mathématique de l'Université de Perm **2**, 105–123 (1919)
5. Besicovitch, A.S.: On Kakeya's problem and a similar one. Mathematische Zeitschrift **27**, 312–320 (1928)
6. Billingsley, P.: Hausdorff dimension in probability theory. Illinois J. Math **4**, 187–209 (1960)
7. Bishop, C.J., Peres, Y.: Fractals in Probability and Analysis. Cambridge University Press (2017)
8. Borel, E.: Sur les probabilités dénombrables et leurs applications arithmétiques. Rendiconti del Circolo Matematico di Palermo **27**(1), 247–271 (1909)
9. Bourke, C., Hitchcock, J.M., Vinodchandran, N.V.: Entropy rates and finite-state dimension. Theoretical Computer Science **349**(3), 392–406 (2005)
10. Braverman, M., Cook, S.A.: Computing over the reals: Foundations for scientific computing. Notices of the AMS **53**(3), 318–329 (2006)

11. Cai, J., Hartmanis, J.: On Hausdorff and topological dimensions of the Kolmogorov complexity of the real line. Journal of Computer and Systems Sciences **49**, 605–619 (1994)
12. Cajar, H.: Billingsley Dimension in Probability Spaces, *Lecture Notes in Mathematics*, vol. 892. Springer (1981)
13. Case, A., Lutz, J.H.: Mutual dimension. ACM Transactions on Computation Theory **7**(3), Art. 12, 26 (2015)
14. Case, A., Lutz, J.H.: Mutual dimension and random sequences. Theoretical Computer Science **731**, 68–87 (2018)
15. Cassels, J.W.S.: On a problem of Steinhaus about normal numbers. Colloquium Mathematicum **7**, 95–101 (1959)
16. Conidis, C.: Effective packing dimension of Π^0_1-classes. Proceedings of the American Mathematical Society **136**, 3655–3662 (2008)
17. Couch, P.J., Daniel, B.D., McNicholl, T.H.: Computing space-filling curves. Theory of Computing Systems **50**(2), 370–386 (2012)
18. Cover, T.M., Thomas, J.A.: Elements of Information Theory, 2nd edition. John Wiley & Sons, Inc., New York, N.Y. (2006)
19. Dai, J.J., Lathrop, J.I., Lutz, J.H., Mayordomo, E.: Finite-state dimension. Theoretical Computer Science **310**, 1–33 (2004)
20. Davies, R.O.: Some remarks on the Kakeya problem. Proc. Cambridge Phil. Soc. **69**, 417–421 (1971)
21. Diamondstone, D., Kjos-Hanssen, B.: Members of random closed sets. In: K. Ambos-Spies, B. Löwe, W. Merkle (eds.) CiE, *Lecture Notes in Computer Science*, vol. 5635, pp. 144–153. Springer (2009)
22. Dougherty, R., Lutz, J.H., Mauldin, R.D., Teutsch, J.: Translating the Cantor set by a random real. Transactions of the American Mathematical Society **366**, 3027–3041 (2014)
23. Downey, R.G., Hirschfeldt, D.R.: Algorithmic Randomness and Complexity. *Theory and Applications of Computability*. Springer, New York (2010)
24. Falconer, K.: Dimensions and measures of quasi self-similar sets. Proc. Amer. Math. Soc. **106**, 543–554 (1989)
25. Falconer, K.: Fractal Geometry: Mathematical Foundations and Applications, 3rd edition. John Wiley & Sons (2014)
26. Falconer, K.J.: Sets with large intersection properties. Journal of the London Mathematical Society **49**(2), 267–280 (1994)
27. Feder, M.: Gambling using a finite state machine. IEEE Transactions on Information Theory **37**, 1459–1461 (1991)
28. Federer, H.: Geometric Measure Theory. Springer-Verlag (1969)
29. Gu, X., Lutz, J.H., Mayordomo, E.: Points on computable curves. In: 47th Annual IEEE Symposium on Foundations of Computer Science, pp. 469–474. IEEE Computer Society Press (2006), proceedings of FOCS 2006, Berkeley, CA, October 22–24, 2006
30. Gu, X., Lutz, J.H., Mayordomo, E.: Curves that must be retraced. Information and Computation **209**(6), 992–1006 (2011)
31. Gu, X., Lutz, J.H., Mayordomo, E., Moser, P.: Dimension spectra of random subfractals of self-similar fractals. Annals of Pure and Applied Logic **165**(11), 1707–1726 (2014)
32. Hausdorff, F.: Dimension und äußeres Maß. Math. Ann. **79**, 157–179 (1919)
33. Hitchcock, J., Lutz, J., Terwijn, S.: The arithmetical complexity of dimension and randomness. ACM Transactions on Computational Logic **8** (2007)
34. Hitchcock, J.M.: Correspondence principles for effective dimensions. Theory of Computing Systems **38**, 559–571 (2005)
35. Hitchcock, J.M., Lutz, J.H., Mayordomo, E.: The fractal geometry of complexity classes. SIGACT News Complexity Theory Column **36**, 24–38 (2005)
36. Hitchcock, J.M., Mayordomo, E.: Base invariance of feasible dimension. Inform. Process. Lett. **113**(14-16), 546–551 (2013)
37. Hitchcock, J.M., Pavan, A.: Resource-bounded strong dimension versus resource-bounded category. Information Processing Letters **95**, 377–381 (2005)

38. Hitchcock, J.M., Vinodchandran, N.V.: Dimension, entropy rates, and compression. Journal of Computer and Systems Sciences **72**(4), 760–782 (2006)
39. Jones, P.W.: Rectifiable sets and the traveling salesman problem. Inventions Mathematicae **102**, 1–15 (1990)
40. Kahane, J.P.: Sur la dimension des intersections. In: J.A. Barroso (ed.) Aspects of Mathematics and its Applications, *North-Holland Mathematical Library*, vol. 34, pp. 419–430. Elsevier (1986)
41. Kamo, H., Kawamura, K.: Computability of self-similar sets. Mathematical Logic Quarterly **45**, 23–30 (1999)
42. Katz, N.H., Tao, T.: Some connections between Falconer's distance set conjecture and sets of Furstenburg type. New York Journal of Mathematics **7**, 149–187 (2001)
43. Ko, K.I.: Complexity Theory of Real Functions. *Progress in Theoretical Computer Science*. Birkhäuser, Boston (1991)
44. Kolmogorov, A.N.: Three approaches to the quantitative definition of 'information'. Problems of Information Transmission **1**, 1–7 (1965)
45. Lacombe, D.: Extension de la notion de fonction récursive aux fonctions d'une ou plusieurs variables réelles I. Comptes Rendus Académie des Sciences Paris **240**, 2478–2480 (1955)
46. Lacombe, D.: Extension de la notion de fonction récursive aux fonctions d'une ou plusieurs variables réelles II. Comptes Rendus Académie des Sciences Paris **241**, 13–14 (1955)
47. Levin, L.A.: On the notion of a random sequence. Soviet Mathematics Doklady **14**, 1413–1416 (1973)
48. Levin, L.A.: Laws of information conservation (nongrowth) and aspects of the foundation of probability theory. Problems of Information Transmission **10**, 206–210 (1974)
49. Li, M., Vitányi, P.M.B.: An Introduction to Kolmogorov Complexity and its Applications. Springer-Verlag, Berlin (2008), third Edition.
50. López-Valdés, M., Mayordomo, E.: Dimension is compression. Theory of Computing Systems **52**, 95–112 (2013)
51. Lutz, J.H.: Dimension in complexity classes. In: Proceedings of the 15th Annual IEEE Conference on Computational Complexity, pp. 158–169 (2000)
52. Lutz, J.H.: Dimension in complexity classes. SIAM Journal on Computing **32**(5), 1236–1259 (2003)
53. Lutz, J.H.: The dimensions of individual strings and sequences. Information and Computation **187**(1), 49–79 (2003)
54. Lutz, J.H.: Effective fractal dimensions. Mathematical Logic Quarterly **51**(1), 62–72 (2005)
55. Lutz, J.H., Lutz, N.: Lines missing every random point. Computability **4**(2), 85–102 (2015)
56. Lutz, J.H., Lutz, N.: Algorithmic information, plane Kakeya sets, and conditional dimension. ACM Transactions on Computation Theory **10**(2), Art. 7, 22 (2018)
57. Lutz, J.H., Mayordomo, E.: Dimensions of points in self-similar fractals. SIAM Journal on Computing **38**(3), 1080–1112 (2008)
58. Lutz, J.H., Mayordomo, E.: Computing absolutely normal numbers in nearly linear time. Tech. Rep. arXiv:1611.05911, arXiv.org (2016)
59. Lutz, J.H., Weihrauch, K.: Connectivity properties of dimension level sets. Mathematical Logic Quarterly **54**, 483–491 (2008)
60. Lutz, N.: A note on pointwise dimensions. Tech. Rep. arXiv:1612.05849, arXiv.org (2016)
61. Lutz, N.: Fractal intersections and products via algorithmic dimension. In: K.G. Larsen, H.L. Bodlaender, J.F. Raskin (eds.) 42nd International Symposium on Mathematical Foundations of Computer Science, *LIPIcs. Leibniz Int. Proc. Inform.*, vol. 83, pp. Art. No. 58, 12. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern (2017)
62. Lutz, N., Stull, D.: Dimension spectra of lines. In: J. Kari, F. Manea, I. Petre (eds.) Unveiling Dynamics and Complexity, *Lecture Notes in Computer Science*, vol. 10307, pp. 304–314. Springer, Cham (2017), 13th Conference on Computability in Europe, CiE 2017, Turku, Finland, June 12-16, 2017
63. Lutz, N., Stull, D.M.: Bounding the dimension of points on a line. In: T.V. Gopal, G. Jäger, S. Steila (eds.) Theory and Applications of Models of Computation, *Lecture Notes in Computer Science*, vol. 10185, pp. 425–439. Springer, Cham (2017)

64. Marstrand, J.M.: Some fundamental geometrical properties of plane sets of fractional dimensions. Proceedings of the London Mathematical Society **4**(3), 257–302 (1954)
65. Martin-Löf, P.: The definition of random sequences. Information and Control **9**, 602–619 (1966)
66. Mattila, P.: Hausdorff dimension and capacities of intersections of sets in n-space. Acta Mathematica **152**, 77–105 (1984)
67. Mattila, P.: On the Hausdorff dimension and capacities of intersections. Mathematika **32**, 213–217 (1985)
68. Mayordomo, E.: A Kolmogorov complexity characterization of constructive Hausdorff dimension. Information Processing Letters **84**(1), 1–3 (2002)
69. Mayordomo, E.: Effective fractal dimension in algorithmic information theory. In: S.B. Cooper, B. Löwe, A. Sorbi (eds.) New Computational Paradigms, pp. 259–285. Springer, New York (2008)
70. Mayordomo, E.: Effective Hausdorff dimension in general metric spaces. Theory of Computing Systems **62**(7), 1620–1636 (2018)
71. McNicholl, T.H.: Computing links and accessing arcs. Mathematical Logic Quarterly **59**(1-2), 101–107 (2013)
72. Molter, U., Rela, E.: Furstenberg sets for a fractal set of directions. Proceedings of the American Mathematical Society **140**, 2753–2765 (2012)
73. Moran, P.: Additive functions of intervals and Hausdorff dimension. Proceedings of the Cambridge Philosophical Society **42**, 5–23 (1946)
74. Moschovakis, Y.N.: Descriptive Set Theory. North-Holland, Amsterdam (1980)
75. Nies, A.: Computability and Randomness, *Oxford Logic Guides*, vol. 51. Oxford University Press, New York (2009)
76. Okikiolu, K.: Characterization of subsets of rectifiable curves in $\mathbb{R}^n$. Journal of the London Mathematical Society **46**(2), 336–348 (1992)
77. Reimann, J.: Computability and fractal dimension. Ph.D. thesis. University of Heidelberg (2004)
78. Rettinger, R., Zheng, X.: Points on computable curves of computable lengths. In: Mathematical Foundations of Computer Science 2009, *Lecture Notes in Computer Science*, vol. 5734, pp. 736–743. Springer, Berlin (2009)
79. Ryabko, B.Y.: Coding of combinatorial sources and Hausdorff dimension. Soviet Mathematics Doklady **30**, 219–222 (1984)
80. Ryabko, B.Y.: Noiseless coding of combinatorial sources. Problems of Information Transmission **22**, 170–179 (1986)
81. Ryabko, B.Y.: Algorithmic approach to the prediction problem. Problems of Information Transmission **29**, 186–193 (1993)
82. Ryabko, B.Y.: The complexity and effectiveness of prediction problems. Journal of Complexity **10**, 281–295 (1994)
83. Sagan, H.: Space-Filling Curves. *Universitext*. Springer (1994)
84. Schmidt, W.M.: On normal numbers. Pacific J. Math **10**(2), 661–672 (1960)
85. Schnorr, C.P., Stimm, H.: Endliche Automaten und Zufallsfolgen. Acta Informatica **1**(4), 345–359 (1972)
86. Shannon, C.E.: A mathematical theory of communication. Bell System Technical Journal **27**, 379–423, 623–656 (1948)
87. Shen, A., Uspensky, V.A., Vereshchagin, N.: Kolmogorov Complexity and Algorithmic Randomness. American Mathematical Society (2017)
88. Shen, A., Vereshchagin, N.: Logical operations and Kolmogorov complexity. Theoretical Computer Science **271**(1–2), 125–129 (2002)
89. Staiger, L.: Kolmogorov complexity and Hausdorff dimension. Information and Computation **103**, 159–94 (1993)
90. Staiger, L.: A tight upper bound on Kolmogorov complexity and uniformly optimal prediction. Theory of Computing Systems **31**, 215–29 (1998)

91. Staiger, L.: How much can you win when your adversary is handicapped? In: I. Althöfer, N. Cai, G. Dueck, L. Khachatrian, M.S. Pinsker, A. Sarközyand, I. Wegener, Z. Zhang (eds.) Numbers, Information and Complexity, pp. 403–412. Kluwer (2000)
92. Staiger, L.: The Kolmogorov complexity of infinite words. Theoretical Computer Science **383**, 187–199 (2007)
93. Stein, E.M., Shakarchi, R.: Real Analysis: Measure Theory, Integration, and Hilbert Spaces. *Princeton Lectures in Analysis*. Princeton University Press (2005)
94. Sullivan, D.: Entropy, Hausdorff measures old and new, and limit sets of geometrically finite Kleinian groups. Acta Mathematica **153**, 259–277 (1984)
95. Tao, T.: From rotating needles to stability of waves: emerging connections between combinatorics, analysis, and PDE. Notices Amer. Math. Soc. **48**, 294–303 (2000)
96. Tricot, C.: Two definitions of fractional dimension. Mathematical Proceedings of the Cambridge Philosophical Society **91**, 57–74 (1982)
97. Turetsky, D.: Connectedness properties of dimension level sets. Theoretical Computer Science **412**(29), 3598–3603 (2011)
98. Turing, A.M.: On computable numbers, with an application to the "Entscheidungsproblem". A correction. Proceedings of the London Mathematical Society **43**(2), 544–546 (1938)
99. Wegmann, H.: Über den Dimensionsbegriff in Wahrscheinlichkeitsräumen von P. Billingsley I and II. Z. Wahrscheinlichkeitstheorie verw. Geb. **9**, 216–221 and 222–231 (1968)
100. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)
101. Wolff, T.: Recent work connected with the Kakeya problem. In: Prospects in Mathematics, pp. 129–162. AMS (1999)
102. Zheng, X., Rettinger, R.: Point-separable classes of simple computable planar curves. Log. Methods Comput. Sci. **8**(3), 3:15, 19 (2012)

Part III
Constructivity, Logic, and Descriptive Complexity

Chapter 9
Admissibly Represented Spaces and Qcb-Spaces

Matthias Schröder

Abstract A basic concept of Type Two Theory of Effectivity (TTE) is the notion of an admissibly represented space. Admissibly represented spaces are closely related to qcb-spaces. The latter form a well-behaved subclass of topological spaces. In this chapter we give a survey of basic facts about Type Two Theory of Effectivity, admissibly represented spaces, qcb-spaces and effective qcb-spaces. Moreover, we discuss the relationship of qcb-spaces to other categories relevant to Computable Analysis.

9.1 Introduction

Computable Analysis investigates computability on real numbers and related spaces. Type Two Theory of Effectivity (TTE) constitutes a popular approach to Computable Analysis, providing a rigorous computational framework for non-discrete spaces with cardinality of the continuum (cf. [43, 44]). The basic tools of this framework are representations. A representation equips the objects of a given space with names, giving rise to the concept of a represented space. Computable functions between represented spaces are those which are realized by a computable function on the names. The ensuing category of represented spaces and computable functions enjoys excellent closure properties.

Any represented space is equipped with a natural topology, turning it into a qcb-space. Qcb-spaces form a subclass of topological spaces with a remarkably rich structure. For example it is cartesian closed, hence products and function spaces can be formed.

Admissibility is a notion of topological well-behavedness for representations. The category of admissibly represented spaces and continuously realizable func-

Matthias Schröder
Department of Mathematics, TU Darmstadt, Germany
e-mail: matthias.schroeder@cca-net.de

V. Brattka, P. Hertling (eds.), *Handbook of Computability and Complexity in Analysis*,
Theory and Applications of Computability, https://doi.org/10.1007/978-3-030-59234-9_9

tions is equivalent to the category QCB_0 of qcb-spaces with the T_0-property. In particular, every qcb_0-space can be equipped with an admissible representation. Qcb_0-spaces form exactly the class of topological spaces which can be handled appropriately by the framework of Type Two Theory of Effectivity.

An important subcategory of admissibly represented spaces is the class of effective qcb-spaces. These are qcb-spaces endowed with representations that are admissible in a computable sense. Computable metric spaces endowed with their Cauchy representations yield important examples. The category EffQCB of effective qcb-spaces and computable functions has nice closure properties. Therefore Type Two Theory of Effectivity is applicable to a large class of important spaces occurring in Analysis.

This paper is organised as follows. In Section 9.2 we recall basic notions of Type Two Theory of Effectivity with emphasis on represented spaces and their closure properties. Section 9.3 is devoted to the notion of an admissible representation for general topological spaces. In Section 9.4 we present and discuss qcb-spaces. Furthermore we summarise basic effectivity properties of effective qcb-spaces. Finally, in Section 9.5 we compare the category QCB of qcb-spaces with other categories relevant to Computable Analysis.

9.2 Represented Spaces

We present basic concepts of Type Two Theory of Effectivity (TTE), as developed by K. Weihrauch ([43, 44]). TTE offers a model for rigorous computation for uncountable spaces occurring in Analysis such as the real numbers, separable Banach spaces and Silva spaces. The key idea is the notion of represented spaces and of computable functions between them. We collect important properties of the ensuing categories Rep and $\mathsf{Rep}_{\mathrm{eff}}$ of represented spaces. Finally we mention the notion of multirepresentations generalising ordinary representations.

9.2.1 Representations

In classical recursion theory, computability on a countable space is defined by lifting the well-established notion of a computable natural number function via an appropriate numbering of X. A *numbering* ν of a set X is a partial surjection $\nu\colon \mathbb{N} \dashrightarrow X$. Partiality means that not every number is a representative of an element in X. For uncountable sets X this approach fails, because the large cardinality precludes the existence of a numbering. The basic idea of Type Two Theory of Effectivity (TTE) is to encode the objects of X by infinite words over a finite or countably infinite alphabet Σ, i.e., by elements of the set $\Sigma^{\mathbb{N}} = \{p \,|\, p\colon \mathbb{N} \to \Sigma\}$, and to compute on these names. The corresponding partial surjection $\delta\colon \Sigma^{\mathbb{N}} \dashrightarrow X$ mapping every name $p \in \mathrm{dom}(\delta)$

to the encoded element $\delta(p)$ is called a *representation* of X. If $\delta(p) = x$ then p is called a *name* or *representative* of the element $x \in X$.

Example 9.2.1 (The decimal representation). A well-known representation of the set $\mathbb{R}$ of real numbers is the *decimal representation* ρ_{dec}. As its alphabet Σ_{dec} it uses $\{\mathtt{0},\ldots,\mathtt{9},\mathtt{-},\mathtt{.}\}$. It is defined by

$$\rho_{\mathrm{dec}}(a_{-k}\ldots a_0\,\mathtt{.}\,a_1a_2\ldots) \;:=\; +\sum_{i=-k}^{\infty} a_i/10^i\,,$$

$$\rho_{\mathrm{dec}}(\mathtt{-}a_{-k}\ldots a_0\,\mathtt{.}\,a_1a_2\ldots) := -\sum_{i=-k}^{\infty} a_i/10^i$$

for all $k \in \mathbb{N}$ and all digits $a_{-k}, a_{-k+1}, \ldots \in \{\mathtt{0},\ldots,\mathtt{9}\}$. □

For establishing a simple framework of computability for general spaces it is reasonable to restrict oneself to a single, countably infinite alphabet. We choose the set $\mathbb{N}$ of natural numbers as our alphabet. So we use the *Baire space* $\mathbb{N}^{\mathbb{N}}$ as our space of representatives. One might also use other spaces of representatives. For example, representations of real numbers by elements of $\mathbb{Z}^{\mathbb{N}}$ were used by J. Hauck in [21] and implicitly by A. Grzegorczyk in [20]. Moreover, J. Hauck employed representations based on $\mathbb{Z}^{\mathbb{N}}$ to introduce computability for functions on topological spaces with a recursive basis in [22]. M. Escardó used domains for his approach to higher-order exact real number computation [19].

9.2.2 The Baire Space

The Baire space $\mathbb{N}^{\mathbb{N}}$ is a topological space that has all sequences of natural numbers as its carrier set. Its topology is called *Baire topology* and is generated by the basic opens

$$u\mathbb{N}^{\mathbb{N}} := \{p \in \mathbb{N}^{\mathbb{N}} \mid u \text{ is a prefix of } p\}\,,$$

where u ranges over the set $\mathbb{N}^*$ of all finite strings over $\mathbb{N}$. This base is countable, hence $\mathbb{N}^{\mathbb{N}}$ is a countably based (= second-countable) topological space. Any open set has the form

$$W\mathbb{N}^{\mathbb{N}} := \{p \in \mathbb{N}^{\mathbb{N}} \mid p \text{ has a prefix in } W\}\,,$$

where W is an arbitrary subset of $\mathbb{N}^*$. The Baire space is separable: the elements $u\mathtt{0}^\omega$, where $u\mathtt{0}^\omega$ denotes the sequence with prefix $u \in \mathbb{N}^*$ followed by infinitely many $\mathtt{0}$'s, form a countable dense subset.

The basic open sets $u\mathbb{N}^{\mathbb{N}}$ are not only open, but also closed. Such sets are usually called *clopen* (abbreviating closed-and-open). Topological spaces having a basis consisting of clopen sets are called *zero-dimensional*.

The Baire topology is induced by the metric $d_{\mathbb{N}^{\mathbb{N}}} : \mathbb{N}^{\mathbb{N}} \times \mathbb{N}^{\mathbb{N}} \to \mathbb{R}_{\geq 0}$ defined by

$$d_{\mathbb{N}^{\mathbb{N}}}(p,p) = 0 \;\text{ and }\; d_{\mathbb{N}^{\mathbb{N}}}(p,q) := 2^{-\min\{i \in \mathbb{N} \mid p(i) \neq q(i)\}}$$

for all $p \neq q$ in $\mathbb{N}^{\mathbb{N}}$. Note that topological continuity coincides with ε-δ-continuity for functions between metrisable spaces. The metric $d_{\mathbb{N}^{\mathbb{N}}}$ is complete, meaning that every Cauchy sequence $(p_n)_n$ has a limit $p := \lim_{n\to\infty} p_n$. Therefore $\mathbb{N}^{\mathbb{N}}$ is a separable completely metrisable topological space. Such spaces are referred to as *Polish spaces*.

The Baire space $\mathbb{N}^{\mathbb{N}}$ is homeomorphic to its own binary product $\mathbb{N}^{\mathbb{N}} \times \mathbb{N}^{\mathbb{N}}$ and to the countable product $\prod_{i\in\mathbb{N}} \mathbb{N}^{\mathbb{N}}$ of itself. A pairing function $\langle\cdot,\cdot\rangle\colon \mathbb{N}^{\mathbb{N}} \times \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ for $\mathbb{N}^{\mathbb{N}}$ is given by

$$\langle p,q\rangle(2n) := p(n) \text{ and } \langle p,q\rangle(2n+1) := q(n).$$

We denote the projections of its inverse by $\pi_1, \pi_2\colon \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$. Moreover, a countable tupling function $\langle\cdot\rangle\colon \prod_{i\in\mathbb{N}} \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ is defined by

$$\langle p_0, p_1, p_2, \ldots\rangle(k) := p_{fst(k)}(snd(k)),$$

where $fst, snd\colon \mathbb{N} \to \mathbb{N}$ are the computable projections of a canonical computable pairing function on $\mathbb{N}$. We denote by $\pi_{\infty,i}\colon \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ the i-th projection of the inverse. The pairing function $\langle\cdot,\cdot\rangle$ on $\mathbb{N}^{\mathbb{N}}$ as well as the projection functions $\pi_1, \pi_2, \pi_{\infty,0}, \pi_{\infty,1}, \ldots$ are computable in the sense of Subsection 9.2.3.

9.2.3 Computability on the Baire Space

Computability for functions on the Baire space $\mathbb{N}^{\mathbb{N}}$ can be defined either by computable monotone string functions or by Type-2 machines. We present the first option.

A string function $h\colon \mathbb{N}^* \to \mathbb{N}^*$ is called *monotone* if it is monotone with respect to the prefix order $\sqsubseteq$ on the set $\mathbb{N}^\infty := \mathbb{N}^* \cup \mathbb{N}^{\mathbb{N}}$ of finite and countably infinite strings over $\mathbb{N}$. A monotone string function $h\colon \mathbb{N}^* \to \mathbb{N}^*$ generates a partial function h^ω on the Baire space defined by

$$\begin{aligned}\mathrm{dom}(h^\omega) &:= \big\{p \in \mathbb{N}^{\mathbb{N}} \,\big|\, \{h(u) \,|\, u \text{ is a prefix of } p\} \text{ is infinite}\big\}\\ h^\omega(p) &:= \sup\nolimits_{\sqsubseteq} \big\{h(u) \,\big|\, u \text{ is a prefix of } p\big\}\end{aligned}$$

for all $p \in \mathrm{dom}(h^\omega)$. For example, the identity function on $\mathbb{N}^*$ generates the identity function on $\mathbb{N}^{\mathbb{N}}$.

A partial function $g\colon \mathbb{N}^{\mathbb{N}} \dashrightarrow \mathbb{N}^{\mathbb{N}}$ is called *computable* if there is a computable monotone function $h\colon \mathbb{N}^* \to \mathbb{N}^*$ generating g, i.e., $g = h^\omega$. Computability of multivariate functions on $\mathbb{N}^{\mathbb{N}}$ is defined analogously. A *computable element* of $\mathbb{N}^{\mathbb{N}}$ is just some $p \in \mathbb{N}^{\mathbb{N}}$ which is computable as a natural number function in the sense of discrete computability theory.

Example 9.2.2. The pairing function $\langle\cdot,\cdot\rangle$ and the projection functions $\pi_1, \pi_2, \pi_{\infty,i}$ from Subsection 9.2.2 are computable. □

Computable functions on $\mathbb{N}^{\mathbb{N}}$ are closed under composition.

Proposition 9.2.3 (Composition preserves computability).

1) *Let* $f\colon (\mathbb{N}^{\mathbb{N}})^k \dashrightarrow \mathbb{N}^{\mathbb{N}}$ *and* $g_1,\ldots,g_k\colon (\mathbb{N}^{\mathbb{N}})^\ell \dashrightarrow \mathbb{N}^{\mathbb{N}}$ *be computable functions. Then the composition* $f\circ(g_1\times\ldots\times g_k)$ *is computable.*
2) *Let* g *be a partial computable function on* $\mathbb{N}^{\mathbb{N}}$ *and let* $p\in\mathrm{dom}(g)$ *be a computable element of* $\mathbb{N}^{\mathbb{N}}$*. Then* $g(p)$ *is a computable element of* $\mathbb{N}^{\mathbb{N}}$*.*

The following key observation links computability theory with topology: any computable function on the Baire space is continuous.

Proposition 9.2.4 (Computability implies continuity).
Any computable function $g\colon \mathbb{N}^{\mathbb{N}}\times\ldots\times\mathbb{N}^{\mathbb{N}} \dashrightarrow \mathbb{N}^{\mathbb{N}}$ *is topologically continuous.*

Proof. We restrict ourselves to the unary case. So there is a computable monotone function $h\colon \mathbb{N}^*\to\mathbb{N}^*$ such that $h^\omega = g$. Let $w\mathbb{N}^{\mathbb{N}}$ be a basic open (see Subsection 9.2.2) and let $p\in\mathrm{dom}(g)$ with $p\in w\mathbb{N}^{\mathbb{N}}$. Then there is some $u\in\mathbb{N}^*$ such that $u\sqsubseteq p$ and $w\sqsubseteq h(u)\sqsubseteq g(p)$. This implies that for all $q\in\mathrm{dom}(g)$ with $u\sqsubseteq q$ we have $g(q)\in w\mathbb{N}^{\mathbb{N}}$. Therefore $p\in u\mathbb{N}^{\mathbb{N}}\cap\mathrm{dom}(g)\subseteq g^{-1}[w\mathbb{N}^{\mathbb{N}}]$. We conclude that g is topologically continuous. □

Example 9.2.5 (An incomputable function on $\mathbb{N}^{\mathbb{N}}$*).*
The test function *is_zero*$\colon \mathbb{N}^{\mathbb{N}}\to\mathbb{N}^{\mathbb{N}}$ defined by

$$is\_zero(p) := \begin{cases} 0^\omega \text{ if } p = 0^\omega \\ 1^\omega \text{ otherwise} \end{cases}$$

is incomputable because it is discontinuous. □

The set $\mathrm{F}^{\omega\omega}$ of all partial continuous functions on $\mathbb{N}^{\mathbb{N}}$ with a G_δ-domain can be endowed with an "effective" representation $\bar\eta$ which satisfies the computable utm-Theorem, the computable smn-Theorem and the continuous smn-Theorem.

Proposition 9.2.6. *There is a representation* $\bar\eta\colon \mathbb{N}^{\mathbb{N}}\to\mathrm{F}^{\omega\omega}$ *such that:*

1) *The evaluation function* $(p,q)\mapsto\bar\eta(p)(q)$ *is computable.*
2) *For every computable function* $g:\mathbb{N}^{\mathbb{N}}\times\mathbb{N}^{\mathbb{N}} \dashrightarrow \mathbb{N}^{\mathbb{N}}$ *there is a computable function* $s:\mathbb{N}^{\mathbb{N}}\to\mathbb{N}^{\mathbb{N}}$ *satisfying* $\bar\eta(s(p))(q) = g(p,q)$ *for all* $p,q\in\mathbb{N}^{\mathbb{N}}$*.*
3) *For every partial continuous function* $g\colon \mathbb{N}^{\mathbb{N}}\times\mathbb{N}^{\mathbb{N}} \dashrightarrow \mathbb{N}^{\mathbb{N}}$ *there is a continuous function* $s\colon \mathbb{N}^{\mathbb{N}}\to\mathbb{N}^{\mathbb{N}}$ *satisfying* $\bar\eta(s(p))(q) = g(p,q)$ *for all* $(p,q)\in\mathrm{dom}(g)$*.*

Explicit constructions of $\bar\eta$ can be found, e.g., in [4, 29, 33, 43]. The application $p\odot q := \bar\eta(p)(q)$ induces on $\mathbb{N}^{\mathbb{N}}$ the structure of a partial combinatory algebra (pca), cf. [3, 29]. The pair $(\mathbb{N}^{\mathbb{N}},\odot)$ is often referred to as Kleene's second algebra K_2.

9.2.4 Represented Spaces

A *represented space* is a pair $\mathbf{X} = (X, \delta)$ such that X is a set and δ is a Baire space representation of X, i.e., a partial surjection from $\mathbb{N}^{\mathbb{N}}$ onto X. The name 'represented space' was coined by V. Brattka in [6].

9.2.5 Computable Elements of Represented Spaces

An element x of a represented space $\mathbf{X} = (X, \delta)$ is called *computable* (or δ-*computable*) if it has a computable name $p \in \mathbb{N}^{\mathbb{N}}$. Similarly, a *computable sequence* $(x_i)_i$ in $\mathbf{X}$ is a sequence of points such that there exists a computable $p \in \mathbb{N}^{\mathbb{N}}$ such that $\delta(\pi_i^\infty(p)) = x_i$ for all $i \in \mathbb{N}$. The reader should be warned that a sequence of computable elements need not be a computable sequence of elements.

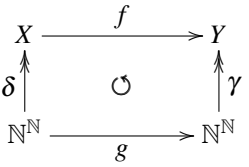
Example 9.2.7 (Computable real numbers).
The real numbers which are ρ_{dec}-computable are exactly those numbers $x \in \mathbb{R}$ for which there is a computable sequence $(z_k)_k$ of integers satisfying $|\frac{z_k}{k+1} - x| \leq \frac{1}{k+1}$ for all $k \in \mathbb{N}$. The latter is the common definition of a computable real number as introduced by A. Grzegorczyk (see [20]). Rational numbers and $\sqrt{2}, \pi, \exp(1)$ are computable real numbers, whereas for any non-decidable subset M of natural numbers, $\sum_{i \in M} 4^{-i}$ is an incomputable real number. □

Example 9.2.8 (Computable sequences of real numbers).
A sequence $(x_i)_i$ of real numbers is called *computable* if, and only if, there is a computable double sequence $(q_{i,k})_{i,k}$ of rational numbers such that $|q_{i,k} - x_i| \leq 2^{-k}$ for all $i, k \in \mathbb{N}$. This is equivalent to saying that $(x_i)_i$ is a computable sequence of the represented space $(\mathbb{R}, \rho_{\mathrm{dec}})$. □

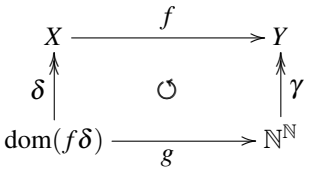
9.2.6 Computable Realizability

A function f between two represented spaces $\mathbf{X} = (X, \delta)$ and $\mathbf{Y} = (Y, \gamma)$ is called *computable* if there is a partial computable function $g\colon \mathbb{N}^{\mathbb{N}} \dashrightarrow \mathbb{N}^{\mathbb{N}}$ *realizing* f, meaning that $f\delta(p) = \gamma g(p)$ holds for all $p \in \mathrm{dom}(f\delta)$. This notion of computability is also called (δ, γ)-*computability* or *computable realizability* w.r.t. δ and γ.

That fact that a *realizer* g realizes a function f is usually visualised by a commuting diagram like the following:



We remark that this diagram is slightly imprecise, since it does not capture the situation for $p \notin \mathrm{dom}(f\delta)$: in particular g is not required to diverge on names $p \in \mathrm{dom}(\delta) \setminus \mathrm{dom}(f\delta)$. So a more precise commuting diagram looks like:

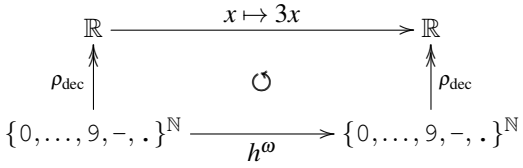


Computability of partial multivariate functions $f\colon \mathbf{X}_1 \times \ldots \times \mathbf{X}_k \dashrightarrow \mathbf{Y}$ between represented spaces is defined via the existence of a computable realizer $g\colon (\mathbb{N}^{\mathbb{N}})^k \dashrightarrow \mathbb{N}^{\mathbb{N}}$.

By Proposition 9.2.4, any computable function between represented spaces maps computable elements in the domain to computable elements in the image; moreover composition of functions between represented spaces preserves computability. Clearly the identity function on a represented space is computable. So the class of represented spaces and the total computable functions between them forms a category which we denote by $\mathsf{Rep}_{\mathrm{eff}}$.

Example 9.2.9 (Computability with respect to the decimal representation).
Real multiplication by 2 is computable with respect to the decimal representation ρ_{dec}. By contrast, multiplication by 3 is *not* computable w.r.t. ρ_{dec}. This was already observed by A. Turing in [42]. We present the simple, well-known proof. To fit into our framework, we identify the symbols – and . with the numbers ten and eleven, respectively, so that we can view Σ_{dec} as a subset of $\mathbb{N}$ and ρ_{dec} as a Baire space representation.

Suppose for contradiction that h is a monotone computable function such that h^ω realizes multiplication by 3 with respect to ρ_{dec}:



The sequence $p := \mathtt{0.3333\ldots}$ is a name of $\frac{1}{3}$. Hence $h^\omega(p)$ must be equal to $\mathtt{0.999\ldots}$ or to $\mathtt{1.000\ldots}$; let us assume the first case. By the continuity of h^ω, there is a finite prefix $u = \mathtt{0.33\ldots3}$ of p such that $\mathtt{0.33\ldots3} \sqsubseteq q$ implies $\mathtt{0.9} \sqsubseteq h^\omega(q)$. But then h^ω works incorrectly on the name $p' := u\mathtt{4000\ldots} = \mathtt{0.33\ldots34000\ldots}$, which represents a real number strictly larger than $\frac{1}{3}$. In the other case we obtain a similar contradiction. We conclude that multiplication by 3 cannot be computable on $(\mathbb{R}, \rho_{\mathrm{dec}})$. □

We obtain that the basic arithmetic operations $+, -, *, \div$ on the reals are incomputable with respect to the decimal representation.

9.2.7 The Cauchy Representation of the Real Numbers

We present a representation of the real numbers which induces a reasonable notion of computability. It is known as the *Cauchy representation* for the reals. The Cauchy representation $\rho_{\mathbb{R}}$ is defined by

$$\rho_{\mathbb{R}}(p) = x \;:\Longleftrightarrow\; \forall k \in \mathbb{N}.|\nu_{\mathbb{Q}}(p(k)) - x| \leq 2^{-k},$$

where $\nu_{\mathbb{Q}}$ is a canonical numbering[1] of the rational numbers $\mathbb{Q}$. So a name of a real number is essentially a rapidly converging sequence of rationals.

We present an incomplete list of functions which are computable with respect to the Cauchy representation.

Example 9.2.10 (Computable real functions).
The following (total or partial) real functions are computable with respect to the Cauchy representation on their natural domain.

1) $+, -, *, \div$
2) $\sqrt{x}$, $\sqrt[y]{x}$, exp, log
3) sin, cos, tan, arcsin, arccos, arctan
4) $x \mapsto \int_0^x f\mathrm{d}t$ for any computable function $f\colon (\mathbb{R}, \rho_{\mathbb{R}}) \to (\mathbb{R}, \rho_{\mathbb{R}})$. □

These facts induce us to choose the following as the notion of computability on the real numbers.

Definition 9.2.11. A total or partial real function $f\colon \mathbb{R}^k \dashrightarrow \mathbb{R}$ is called *computable* if it is computable with respect to the Cauchy representation $\rho_{\mathbb{R}}$. A real number x is *computable* if it has a computable name p with $\rho_{\mathbb{R}}(p) = x$.

This notion of computability is essentially the same as the one considered by A. Grzegorczyk in [20], at least for functions with well-behaved domain. More information about real computability can be found, e.g., in [6, 7, 10, 11, 44].

9.2.8 The Signed-Digit Representation

We present an alternative representation of reals inducing the same notion of computability as the Cauchy representation. It is called the *signed-digit representation*.

A straightforward version of the signed-digit representation uses the zero-dimensional Polish space $\mathbb{Z} \times \{\bar{1}, 0, 1\}^{\mathbb{N}}$ as the space of representatives instead of the Baire space $\mathbb{N}^{\mathbb{N}}$, where the symbol $\bar{1}$ stands for the negative digit -1. It is defined by

[1] A canonical numbering of $\mathbb{Q}$ can defined by $\nu_{\mathbb{Q}}(n) := \frac{\nu_{\mathbb{Z}}(fst(n))}{1+snd(n)}$, where fst, snd are the computable projections of a computable pairing function for $\mathbb{N}$ and $\nu_{\mathbb{Z}}$ is a canonical numbering of the integers like the one in Subsection 9.2.8.

$$\widetilde{\rho_{\mathrm{sd}}}(z,p) := z + \sum_{i\in\mathbb{N}} \frac{p(i)}{2^{i+1}}$$

for all integers $z \in \mathbb{Z}$ and all $p \in \{\bar{1},0,1\}^{\mathbb{N}}$. The trick of using the negative digit $\bar{1}$ guarantees that the real numbers are encoded in a reasonable way.

We modify $\widetilde{\rho_{\mathrm{sd}}}$ to obtain a version of the signed-digit representation that fits better into our framework, which uses the Baire space as the fixed representing space. We employ the effective bijective numbering of the set $\mathbb{Z}$ of integers given by

$$\nu_{\mathbb{Z}}(n) := \begin{cases} \frac{n+1}{2} & \text{if } n \text{ is odd} \\ -\frac{n}{2} & \text{otherwise.} \end{cases}$$

So the sequence $(\nu_{\mathbb{Z}}(n))_n$ starts with the initial segment $0,1,-1,2,-2,3,-3$. Now we can define our version of the signed-digit representation $\rho_{\mathrm{sd}} \colon \mathbb{N}^{\mathbb{N}} \dashrightarrow \mathbb{R}$ over the Baire space by

$$\mathrm{dom}(\rho_{\mathrm{sd}}) := \left\{ p \in \mathbb{N}^{\mathbb{N}} \,\middle|\, p(i) \in \{0,1,2\} \text{ for all } i \geq 1 \right\},$$
$$\rho_{\mathrm{sd}}(p) := \nu_{\mathbb{Z}}(p(0)) + \sum_{i\geq 1} \frac{\nu_{\mathbb{Z}}(p(i))}{2^i}.$$

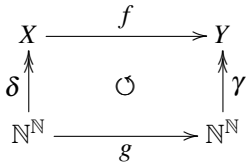
Recall that 2 encodes the negative digit -1.

While inducing the same computability notion as the Cauchy representation (cf. Example 9.2.14), the signed-digit representation has the advantage over the Cauchy representation from Subsection 9.2.7 of being *proper*. This means that the preimage of any compact subset of $\mathbb{R}$ under ρ_{sd} is compact (cf. [36, 44, 45]). This property makes the signed-digit representation suitable for Type Two Complexity Theory.

9.2.9 Continuous Realizability

In Subsection 9.2.3 we have seen that every computable function on the Baire space is continuous with respect to the Baire topology. Hence the notion of a continuously realizable function plays an important role in investigating the question of which representations of a space lead to a reasonable computability notion on that space.

A partial function f between two represented spaces $\mathbf{X} = (X,\delta)$ and $\mathbf{Y} = (Y,\gamma)$ is called *continuously realizable* (for short *continuous*) if there is a partial continuous function g on the Baire space realizing f, meaning that $\gamma g(p) = f\delta(p)$ holds for all $p \in \mathrm{dom}(f\delta)$:



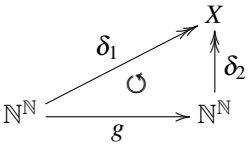
In this situation f is also called (δ,γ)-*continuous* or *continuously realizable* with respect to δ and γ. Continuous realizability of multivariate functions is defined similarly. Since every computable function on $\mathbb{N}^{\mathbb{N}}$ is topologically continuous, we obtain a new instance of the slogan: *Computability implies Continuity*.

Proposition 9.2.12. *Any computable function between represented spaces is continuous.*

Hence continuous realizability can be viewed as a topological generalisation of computable realizability. Computability with respect to an oracle is equivalent to continuous realizability.

9.2.10 Reducibility and Equivalence of Representations

Given two representations δ_1 and δ_2 of the same set X we say that δ_1 is *computably reducible to* (or *computably translatable into*) δ_2 if there is a computable function g on the Baire space translating δ_1 into δ_2 in the sense that the diagram



commutes, meaning $\delta_1(p) = \delta_2(g(p))$ for all $p \in \mathrm{dom}(\delta_1)$. In this situation we write $\delta_1 \leq_{\mathrm{cp}} \delta_2$. If $\delta_1 \leq_{\mathrm{cp}} \delta_2 \leq_{\mathrm{cp}} \delta_1$, then δ_1 and δ_2 are called *computably equivalent*, which is written as $\delta_1 \equiv_{\mathrm{cp}} \delta_2$. Some authors omit the subscript "cp". Topological reducibility (in symbols $\delta_1 \leq_{\mathrm{t}} \delta_2$) and topological equivalence (in symbols $\delta_1 \equiv_{\mathrm{t}} \delta_2$) is defined analogously by means of *continuous* rather than *computable* translator functions. Topological reducibility (equivalence) is also known as *continuous reducibility (equivalence)*. Obviously, $\leq_{\mathrm{cp}}, \leq_{\mathrm{t}}, \equiv_{\mathrm{cp}}, \equiv_{\mathrm{t}}$ are reflexive and transitive. Computably equivalent representations induce the same notion of computability.

Proposition 9.2.13. *Let $\delta_1, \delta_2 \colon \mathbb{N}^{\mathbb{N}} \dashrightarrow X$ and $\gamma_1, \gamma_2 \colon \mathbb{N}^{\mathbb{N}} \dashrightarrow Y$ be representations. Let $f \colon X \dashrightarrow Y$ be a partial function.*

1) *If $\delta_1 \equiv_{\mathrm{cp}} \delta_2$ and $\gamma_1 \equiv_{\mathrm{cp}} \gamma_2$, then f is (δ_1,γ_1)-computable if, and only if, f is (δ_2,γ_2)-computable.*
2) *If $\delta_1 \equiv_{\mathrm{t}} \delta_2$ and $\gamma_1 \equiv_{\mathrm{t}} \gamma_2$, then f is (δ_1,γ_1)-continuous if, and only if, f is (δ_2,γ_2)-continuous.*
3) *If $\delta_1 \equiv_{\mathrm{cp}} \delta_2$, then $x \in X$ is δ_1-computable if, and only if, x is δ_2-computable.*
4) *If $\delta_1 \leq_{\mathrm{cp}} \delta_2$, $\gamma_1 \leq_{\mathrm{cp}} \gamma_2$ and f is (δ_2,γ_1)-computable, then f is (δ_1,γ_2)-computable.*
5) *If $\delta_1 \leq_{\mathrm{t}} \delta_2$, $\gamma_1 \leq_{\mathrm{t}} \gamma_2$ and f is (δ_2,γ_1)-continuous, then f is (δ_1,γ_2)-continuous.*
6) *If $\delta_1 \leq_{\mathrm{cp}} \delta_2$ and $x \in X$ is δ_1-computable, then x is δ_2-computable.*

Example 9.2.14 (Comparison of representations of the reals).
The decimal representation ρ_{dec} is computably translatable into the Cauchy representation $\rho_{\mathbb{R}}$, but $\rho_{\mathbb{R}}$ is not even continuously translatable back to ρ_{dec}. Nevertheless, computability of a real number with respect to ρ_{dec} is equivalent to its computability with respect to the Cauchy representation $\rho_{\mathbb{R}}$. The Cauchy representation is computably equivalent to the signed-digit representation ρ_{sd}. So ρ_{sd} induces the same notion of computability on the Euclidean space as $\rho_{\mathbb{R}}$ by Proposition 9.2.13. □

9.2.11 The Category of Represented Spaces

The category of represented spaces comes in two versions, a continuous one and an effective one. Both have as objects all represented spaces (X,δ). The continuous version, Rep, has as morphisms all total continuously realizable functions between represented spaces. The effective category $\mathsf{Rep}_{\mathrm{eff}}$ only uses all total computable functions as morphisms. The category Rep is essentially equal to the category $\mathsf{Mod}(\mathbb{N}^{\mathbb{N}})$ of modest sets over the Baire space (cf. [4]), whereas $\mathsf{Rep}_{\mathrm{eff}}$ corresponds to $\mathsf{Mod}(\mathbb{N}^{\mathbb{N}},\mathbb{N}^{\mathbb{N}}_{\sharp})$; see A. Bauer's thesis [3] for further information.

9.2.12 Closure Properties of Represented Spaces

The category Rep as well as its subcategory $\mathsf{Rep}_{\mathrm{eff}}$ are known to be cartesian closed (even locally cartesian closed, cf. A. Bauer's thesis [3]). We describe how to canonically construct a binary product $\mathbf{X}\times\mathbf{Y}$ and a function space $\mathbf{Y}^{\mathbf{X}}$ in Rep and identically in $\mathsf{Rep}_{\mathrm{eff}}$.

Let $\mathbf{X}=(X,\delta)$ and $\mathbf{Y}=(Y,\gamma)$ be represented spaces. The carrier set of the binary product $\mathbf{X}\times\mathbf{Y}$ is just the cartesian product $X\times Y$. The construction of the product representation $\delta\boxtimes\gamma$ employs the computable bijection $\langle\cdot,\cdot\rangle\colon\mathbb{N}^{\mathbb{N}}\times\mathbb{N}^{\mathbb{N}}\to\mathbb{N}^{\mathbb{N}}$ and its computable projections $\pi_1,\pi_2\colon\mathbb{N}^{\mathbb{N}}\to\mathbb{N}^{\mathbb{N}}$ from Subsection 9.2.2. One defines $\delta\boxtimes\gamma\colon\mathbb{N}^{\mathbb{N}}\dashrightarrow X\times Y$ by

$$\delta\boxtimes\gamma\langle p,q\rangle:=\big(\delta(p),\gamma(q)\big),$$

which is a useful short form for

$$\begin{aligned}\mathrm{dom}(\delta\boxtimes\gamma)&:=\{r\in\mathbb{N}^{\mathbb{N}}\,|\,\pi_1(r)\in\mathrm{dom}(\delta),\ \pi_2(r)\in\mathrm{dom}(\gamma)\},\\ \delta\boxtimes\gamma(r)&:=\big(\delta(\pi_1(r)),\gamma(\pi_2(r))\big)\quad\text{for } r\in\mathrm{dom}(\delta\boxtimes\gamma).\end{aligned}$$

Another familiar notation for $\delta\boxtimes\gamma$ is $[\delta,\gamma]$.

The carrier set of the exponential $\mathbf{Y}^{\mathbf{X}}$ in Rep and in $\mathsf{Rep}_{\mathrm{eff}}$ consists of the set $C(\mathbf{X},\mathbf{Y})$ of all total continuously realizable functions $f\colon\mathbf{X}\to\mathbf{Y}$. Its canonical function space representation $[\delta\to\gamma]$ is constructed with the help of the effective rep-

resentation $\bar{\eta}$ of all partial continuous endofunctions on $\mathbb{N}^{\mathbb{N}}$ with G_δ-domain from Proposition 9.2.6. For all $p \in \mathbb{N}^{\mathbb{N}}$ and total continuous functions $f\colon \mathbf{X} \to \mathbf{Y}$ one defines

$$[\delta \to \gamma](p) = f \ :\Longleftrightarrow\ \text{the function } \bar{\eta}(p) \text{ realizes } f.$$

The utm-Theorem for $\bar{\eta}$ ensures that the evaluation function

$$eval\colon \mathbf{Y}^{\mathbf{X}} \times \mathbf{X} \to \mathbf{Y}, \quad eval(f,x) := f(x)$$

is computable. The computable smn-Theorem guarantees currying of computable functions: for any computable $f\colon \mathbf{Z} \times \mathbf{X} \to \mathbf{Y}$ the function $curry(f)\colon \mathbf{Z} \to \mathbf{Y}^{\mathbf{X}}$ defined by

$$\big(curry(f)(z)\big)(x) := f(z,x) \ \text{ for all } z \in \mathbf{Z} \text{ and } x \in X$$

is computable from $\mathbf{Z}$ to $\mathbf{Y}^{\mathbf{X}}$; in particular $curry(f)(z)$ is indeed an element of $C(\mathbf{X},\mathbf{Y})$. Note that $curry(f)(z)$ need not be a computable function from $\mathbf{X}$ to $\mathbf{Y}$, unless z is a computable element of $\mathbf{Z}$. Likewise, the continuous smn-Theorem ensures that *curry* maps continuously realizable functions to continuously realizable functions. Hence $\mathbf{Y}^{\mathbf{X}}$ defined by $(C(\mathbf{X},\mathbf{Y}),[\delta \to \gamma])$ is an exponential to $\mathbf{X}$ and $\mathbf{Y}$ both in Rep and in $\mathsf{Rep}_{\mathrm{eff}}$. A total function h from $\mathbf{X}$ to $\mathbf{Y}$ is computable if, and only if, h is a computable element of $\mathbf{Y}^{\mathbf{X}}$. We emphasise that the exponential of $\mathbf{X}$ and $\mathbf{Y}$ in $\mathsf{Rep}_{\mathrm{eff}}$ coincides with the one in Rep, so that its underlying set may contain incomputable (albeit continuously realizable) functions.

In contrast to its subcategory $\mathsf{Rep}_{\mathrm{eff}}$, the category Rep has also countable products and countable coproducts. Let $((X_i,\delta_i))_{i\in\mathbb{N}}$ be a sequence of represented spaces. One uses the computable projections $\pi_{\infty,i}$ of the countable tupling function for $\mathbb{N}^{\mathbb{N}}$ from Subsection 9.2.2 and defines representations $\boxtimes_{i\in\mathbb{N}}\delta_i$ for the cartesian product $\prod_{i\in\mathbb{N}} X_i$ and $\boxplus_{i\in\mathbb{N}}\delta_i$ for the disjoint sum $\bigcup_{i\in\mathbb{N}}(\{i\}\times X_i)$ by

$$\begin{aligned}
(\bigboxtimes_{i\in\mathbb{N}}\delta_i)(p) &:= \big(\delta_0(\pi_{\infty,0}(p)),\delta_1(\pi_{\infty,1}(p)),\delta_2(\pi_{\infty,2}(p)),\dots\big),\\
(\bigboxplus_{i\in\mathbb{N}}\delta_i)(p) &:= \big(p(0),\delta_{p(0)}(p^{\geq 1})\big),
\end{aligned}$$

where $p^{\geq 1}(i) := p(i+1)$. The ensuing represented spaces form a countable product and a countable coproduct for $((X_i,\delta_i))_{i\in\mathbb{N}}$ in Rep, but not necessarily in $\mathsf{Rep}_{\mathrm{eff}}$. The represented space $(\{1\}\times X \cup \{2\}\times Y, \delta \boxplus \gamma)$ is a binary coproduct for $\mathbf{X}$ and $\mathbf{Y}$ in both categories, where the binary coproduct representation $\delta \boxplus \gamma$ is defined analogously to the countable coproduct representation.

We mention the useful notion of the *meet* of represented spaces $((X_i,\delta_i))_i$. It has the intersection $\bigcap_{i\in\mathbb{N}} X_i$ as carrier set and the meet representation $\bigwedge_{i\in\mathbb{N}}\delta_i$ is defined by

$$(\bigwedge_{i\in\mathbb{N}}\delta_i)(p) = x \ :\Longleftrightarrow\ \forall i \in \mathbb{N}.\, \delta_i(\pi_i^\infty(p)) = x.$$

The representation $\delta \wedge \gamma$ of the binary meet $\mathbf{X} \wedge \mathbf{Y}$ is defined analogously.

Any subset M of a represented space $\mathbf{X} = (X,\delta)$ has the corestriction $\delta|^M$ as natural representation which has $\delta^{-1}[M]$ as domain and maps $p \in \delta^{-1}[M]$ to $\delta(p) \in$

M. This entails that both Rep and $\mathsf{Rep}_{\mathrm{eff}}$ have equalisers and thus all finite limits. Moreover, Rep has all countable limits because it is countably cartesian.

9.2.13 Multivalued Functions

A *multivalued function* (or *operation* or *problem*) is a triple $F = (X, Y, \mathrm{Graph}(F))$, where $\mathrm{Graph}(F)$ is a relation between sets X and Y, written as $F : X \rightrightarrows Y$. The set X is called the *source* and Y the *target* of F. The intuition behind this definition is that the elements of $F[x] := \{y \in Y \mid (x,y) \in F\}$ are viewed as the legitimate results for an input x under the multivalued function F. If the domain $\mathrm{dom}(F) := \{x \in X \mid F[x] \neq \emptyset\}$ of F is equal to X, then F is called *total*. Ordinary total or partial functions $f : X \dashrightarrow Y$ can be seen as multivalued functions such that the image $f[x]$ is a singleton for every x in the domain $\mathrm{dom}(f)$ of f.

A multivalued function F between represented space $\mathbf{X} = (X, \delta)$ and $\mathbf{Y} = (Y, \gamma)$ is called *computable* (or *computably realizable* or (δ, γ)*-computable*) if there is a computable function g on $\mathbb{N}^{\mathbb{N}}$ such that

$$\gamma(g(p)) \in F[\delta(p)] \quad \text{for every } p \in \mathrm{dom}(\delta) \text{ with } \delta(p) \in \mathrm{dom}(F).$$

So a realizer g for F picks a possible result $y \in F[x]$ for an input x which may heavily depend on the input name $p \in \delta^{-1}[x]$. From the point of view of logic, a computable realizer for F can be interpreted as a computable realizer of the $\forall\exists$-statement

$$\forall x \in \mathrm{dom}(F).\exists y \in Y.\, (x,y) \in \mathrm{Graph}(F)\,.$$

Similarly, F is called *continuously realizable* (for short *continuous*) if it has a continuous realizer g satisfying the above formula.

Example 9.2.15 (Unsharp tests).
A powerful computable multivalued function is the finite-precision test $T : \mathbb{R}^3 \rightrightarrows \mathbb{N}$ defined by

$$T[x,y,\varepsilon] := (x <_\varepsilon y) := \begin{cases} \{1\} & \text{if } x \leq y - \varepsilon \\ \{0,1\} & \text{if } y - \varepsilon < x < y \\ \{0\} & \text{if } x \geq y \end{cases}$$

for all $x, y, \varepsilon \in \mathbb{R}$ with $\varepsilon > 0$. This $(\rho_{\mathbb{R}} \boxtimes \rho_{\mathbb{R}} \boxtimes \rho_{\mathbb{R}}, \rho_{\mathbb{N}})$-computable test (where $\rho_{\mathbb{N}}(p) := p(0)$) is a basic operation in P. Hertling's and V. Brattka's feasible real random access machine (cf. [10]). By contrast, the ordinary precise test "x < y ?" is incomputable because it is discontinuous.

In his package iRRAM for exact real number computation (cf. [31]), N. Müller has implemented a similar multivalued test *positive* : $\mathbb{R} \times \mathbb{Z} \rightrightarrows \mathbb{N}$. It is defined by

$$\mathit{positive}[x,z] := \begin{cases} \{1\} & \text{if } x > 2^z \\ \{0,1\} & \text{if } -2^z \leq x \leq 2^z \\ \{0\} & \text{if } x < -2^z \end{cases}$$

for all $x \in \mathbb{R}$, $z \in \mathbb{Z}$. The multivalued function *positive* is $(\rho_{\mathbb{R}} \boxtimes \rho_{\mathbb{Z}}, \rho_{\mathbb{N}})$-computable. The unsharp positivity test can often be employed as a computable substitute for the discontinuous, hence incomputable test "x > 0 ?". For a further discussion of unsharp tests we refer to [10, 31]. □

From the computability point of view, the most appropriate definition of composition of two multivalued functions is given by

$$\begin{aligned} \mathrm{dom}(G \circ F) &:= \big\{ x \in \mathrm{dom}(F) \,\big|\, F[x] \subseteq \mathrm{dom}(G) \big\}, \\ G \circ F[x] &:= \big\{ z \,\big|\, \exists y \in F[x].\, z \in G[y] \big\} \text{ for } x \in \mathrm{dom}(G \circ F). \end{aligned}$$

This definition guarantees that the class of computable multivalued functions is closed under composition (see [8]). The seemingly more natural version of composition that has $\{x \in \mathrm{dom}(F) \,|\, F[x] \cap \mathrm{dom}(G) \neq \emptyset\}$ as domain does not preserve computability of multivalued functions.

9.2.14 Multirepresentations

A *multirepresentation* δ of a set X is a partial multivalued function from the Baire space to X which is surjective in the sense that for every $x \in X$ there exists some $p \in \mathbb{N}^{\mathbb{N}}$ such that $x \in \delta[p]$. So multirepresentations generalise representations by allowing any $p \in \mathbb{N}^{\mathbb{N}}$ to be a name of more than one element. Such a pair (X, δ) is called a *multirepresented space*.

A partial function $f\colon X \dashrightarrow Y$ between multirepresented spaces $\mathbf{X} = (X, \delta)$ and $\mathbf{Y} = (Y, \gamma)$ is called *computable* or *computably realizable* or (δ, γ)-*computable* if there is a computable function $g\colon \mathbb{N}^{\mathbb{N}} \dashrightarrow \mathbb{N}^{\mathbb{N}}$ satisfying

$$x \in \delta[p] \;\&\; x \in \mathrm{dom}(f) \implies f(x) \in \gamma[g(p)];$$

similarly for multivariate functions. Continuous realizability w.r.t. multirepresentations is defined via *continuous* realizers rather than computable realizers in the spirit of Subsection 9.2.9. The category MRep of multirepresented spaces (X, δ) and continuously realizable total functions is equivalent to the category $\mathsf{Asm}(\mathbb{N}^{\mathbb{N}})$ of assemblies [29] over the Baire space. Thus MRep is locally cartesian closed, countably complete and countably cocomplete.

There exist constructors for forming canonical multirepresentations for products, coproducts, meets and function spaces generalising those for representations presented in Subsection 9.2.4. Details can be found in [33].

Example 9.2.16 (The space of partial continuous functions).
Let $\mathbf{X} = (X, \delta)$, $\mathbf{Y} = (Y, \gamma)$ and $\mathbf{Z}$ be (multi-)represented spaces. A canonical multirepresentation of the set of partial continuously realizable functions from $\mathbf{X}$ to $\mathbf{Y}$ is defined by

$$f \in [\delta \dashrightarrow \gamma][p] \;:\Longleftrightarrow\; \forall (r,x) \in \mathrm{Graph}(\delta).\big(x \in \mathrm{dom}(f) \implies f(x) \in \gamma[\bar{\eta}(p)(r)]\big),$$

where $\bar{\eta}$ is taken from Proposition 9.2.6. We denote by $\mathbf{P}(\mathbf{X},\mathbf{Y})$ the ensuing multi-represented space. The evaluation function $eval\colon \mathbf{P}(\mathbf{X},\mathbf{Y})\times\mathbf{X} \dashrightarrow \mathbf{Y}$ is computable. Moreover, $[\delta \dashrightarrow \gamma]$ admits currying of the following kind: for all partial computable functions $h\colon \mathbf{Z}\times\mathbf{X} \dashrightarrow \mathbf{Y}$ the total function $curry(h)\colon \mathbf{Z}\to\mathbf{P}(\mathbf{X},\mathbf{Y})$ defined by

$$\mathrm{dom}\big(curry(h)(z)\big) := \{x \,|\, (z,x)\in\mathrm{dom}(h)\},\ \big(curry(h)(z)\big)(x) := h(z,x)$$

is computable as well. □

More information can be found in [33, 47].

9.3 Admissible Representations of Topological Spaces

We introduce and study admissibility as a notion of topological well-behavedness for representations for topological spaces. This notion is a generalisation of a similar notion introduced by C. Kreitz and K. Weihrauch in [26] which applies to representations of second-countable topological spaces.

9.3.1 The Topology of a Represented Space

Any represented space $\mathbf{X}=(X,\delta)$ carries a natural topology, namely the *final topology* or *quotient topology* induced by the partial surjection δ. The final topology is defined as the family of sets

$$\tau_\delta := \big\{U\subseteq X \,\big|\, \delta^{-1}[U] = W\cap\mathrm{dom}(\delta) \text{ for some open subset } W\subseteq\mathbb{N}^{\mathbb{N}}\big\}.$$

It is equal to the finest (= largest) topology on X such that δ is topologically continuous. We denote by $\mathscr{T}(X,\delta)$ the topological space that carries the final topology τ_δ and call it the *associated* space. The following fact is crucial.

Proposition 9.3.1. *Let $\mathbf{X}=(X,\delta)$ and $\mathbf{Y}=(Y,\gamma)$ be represented spaces. Any total continuously realizable function $f\colon X\to Y$ is topologically continuous with respect to the final topologies of the respective representations.*

Hence the operator $\mathscr{T}$ constitutes a forgetful functor $\mathscr{T}\colon \mathsf{Rep}\to\mathsf{Top}$ mapping continuously realizable functions to themselves. In Subsections 9.3.2 and 9.4.1 we will see that $\mathscr{T}(X,\delta)$ is a sequential topological space and even a qcb-space.

Proposition 9.3.2 (Computability implies continuity). *Any total computable function between two represented spaces is topologically continuous with respect to the topologies associated with the represented spaces.*

9.3.2 Sequential Topological Spaces

A *sequential space* (cf. [18, 15, 25]) is a topological space X such that every sequentially closed subset of X is closed in X. A subset A of X is called *sequentially closed* if A contains every limit x of every sequence $(x_n)_n$ in A. Remember that $(x_n)_n$ is defined to converge in a topological space X to an element x (in symbols $(x_n)_n \to_{\mathsf{X}} x$ or shorter $(x_n)_n \to x$) if any open set $U \ni x$ contains $(x_n)_n$ eventually, meaning that there is some $n_0 \in \mathbb{N}$ with $x_n \in U$ for all $n \geq n_0$. The complements of sequentially closed sets are called *sequentially open*.

Basic examples of sequential spaces are metrisable spaces and second-countable spaces. The following fact is crucial.

Example 9.3.3 (Sequential spaces).
For any represented space $\mathbf{X}$, the associated topological space $\mathscr{T}(\mathbf{X})$ is sequential. □

Neither forming the topological product $\mathsf{X} \times_{\mathsf{Top}} \mathsf{Y}$ nor taking the compact-open topology on the family $C(\mathsf{X},\mathsf{Y})$ of total topologically continuous functions from X to Y nor taking a topological subspace preserve the property of being sequential. For example, the compact-open topologies on $C(\mathbb{N}^{\mathbb{N}},\mathbb{N})$ and on $C(\mathbb{R}^{\mathbb{R}},\mathbb{R})$ are not sequential.

A partial function $f\colon \mathsf{X} \dashrightarrow \mathsf{Y}$ between topological spaces is called *sequentially continuous* if f preserves convergent sequences, i.e., $(x_n)_n \to_{\mathsf{X}} x$ implies $(f(x_n))_n \to_{\mathsf{Y}} f(x)$ whenever the sequence $(x_n)_n$ and its limit x are in the domain of f. The map f is called *topologically continuous* if for every open set V in Y there is some open set U in X such that $f^{-1}[V] = U \cap \mathrm{dom}(f)$. Topological continuity implies sequential continuity, but not vice versa. Importantly, sequential continuity coincides with topological continuity for total unary functions between sequential spaces. The reader should be warned that this equivalence does not hold in general for multivariate functions nor for non-total functions.

By Seq we denote the category of sequential topological spaces as objects and total sequentially continuous functions as morphisms. This category is known to be cartesian closed (see [18, 25]). By contrast, the category Top of topological spaces and total topologically continuous functions fails to be cartesian closed.

9.3.3 Sequentialisation of Topological Spaces

The family of all sequentially open sets of a topological space X forms a (possibly) finer topology on the carrier set of X. It is called the *sequentialisation* of the topology of X. The topological space $seq(\mathsf{X})$ carrying this topology is the unique sequential space that has the same convergence relation on sequences as X. It is equal to X if, and only if, X is sequential. The functor *seq* is right-adjoint to the inclusion functor of Seq into the category Top of topological spaces (see [25]). So $seq(\mathsf{X})$ is referred to as the *sequential coreflection* of X.

9.3.4 Topological Admissibility

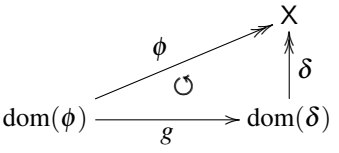
Given represented spaces, the natural question arises of which functions are computable between them. Conversely, one wants to know how to construct appropriate representations that admit computability of a relevant family of functions and thus have certain required effectivity properties.

Since computability w.r.t. given representations implies continuity w.r.t. these representations, it is reasonable to consider two different kinds of effectivity properties, the topological ones and the purely computable ones. Many representations fail to be appropriate simply by having unsuitable topological properties which prevent some interesting functions from being at least continuously realizable. The prime example is the notorious decimal representation ρ_{dec}: an inspection of the proof of Example 9.2.9 shows that multiplication by 3 is in fact not even continuously realizable. So the misbehaviour of the decimal representation is of purely topological nature.

Therefore we turn our attention to continuous realizability and identify those representations of topological spaces which induce an appropriate class of continuously realizable functions. This is done by introducing the property of *admissibility* for representations. It aims at guaranteeing that every topologically continuous function is continuously realizable. Definition 9.3.4 generalises a previous notion of admissibility introduced by C. Kreitz and K. Weihrauch (see [26, 44] and Subsection 9.3.6) which is tailor-made for topological spaces with a countable base.

Definition 9.3.4 (Admissible representations [35]).
A representation $\delta\colon \mathbb{N}^{\mathbb{N}} \dashrightarrow \mathsf{X}$ is called *admissible* for a topological space X if

a) δ is continuous,
b) for every partial continuous function $\phi\colon \mathbb{N}^{\mathbb{N}} \dashrightarrow \mathsf{X}$ there is a continuous function $g\colon \mathbb{N}^{\mathbb{N}} \dashrightarrow \mathbb{N}^{\mathbb{N}}$ satisfying $\phi = \delta \circ g$, i.e., the diagram



commutes.

The condition b) is referred to as the *maximality property* of δ. It postulates that any continuous representation of X can be translated into δ. Any admissible representation of X is maximal in the class of continuous representations of X with respect to continuous reducibility $\leq_{\mathrm{t}}$. Continuous reducibility (= topological reducibility) is defined by: $\phi \leq_{\mathrm{t}} \phi'$ iff there is a continuous $g\colon \mathrm{dom}(\phi) \to \mathbb{N}^{\mathbb{N}}$ with $\phi = \phi' \circ g$.

Proposition 9.3.5. *Let δ be a representation of a topological space X. Then δ is admissible if, and only if, it is $\leq_{\mathrm{t}}$-maximal in the class of continuous representations of X.*

Admissibility ensures the equivalence of mathematical continuity and continuous realizability as desired.

Theorem 9.3.6. *Let δ and γ be admissible representations of topological spaces X and Y, respectively. Then a partial function $f\colon \mathsf{X} \dashrightarrow \mathsf{Y}$ is sequentially continuous if, and only if, $f\colon (\mathsf{X},\delta) \dashrightarrow (\mathsf{Y},\gamma)$ is continuously realizable.*

Recall that topological continuity implies sequential continuity, but not vice versa.

Admissibility is necessary only for the representation of the target space, whereas the source representation is merely required to be a quotient representation (δ is called a *quotient representation* of a topological space X if its final topology is equal to the topology of X). We formulate this fact with the help of the notion of an *admissibly represented space*: this is a represented space $\mathbf{Y}$ such that its representation is admissible for the associated topological space $\mathscr{T}(\mathbf{Y})$. A. Pauly coined this notion in [32] using a different definition which is equivalent to the above definition by Theorem 9.4.8.

Theorem 9.3.7. *Let $\mathbf{X}$ be a represented space and let $\mathbf{Y}$ be an admissibly represented space. Then a total function $f\colon \mathbf{X} \to \mathbf{Y}$ is continuously realizable if, and only if, f is topologically continuous between the associated topological spaces $\mathscr{T}(\mathbf{X})$ and $\mathscr{T}(\mathbf{Y})$.*

For partial functions and for multivariate functions we only get equivalence with sequential continuity.

Theorem 9.3.8. *Let $\mathbf{X}_i$ be represented spaces for $i \in \{1,\ldots,k\}$ and let $\mathbf{Y}$ be an admissibly represented space. Then a partial function $f\colon \mathbf{X}_1 \times \ldots \times \mathbf{X}_k \dashrightarrow \mathbf{Y}$ is continuously realizable if, and only if, f is sequentially continuous between the associated topological spaces $\mathscr{T}(\mathbf{X}_1),\ldots,\mathscr{T}(\mathbf{X}_k)$ and $\mathscr{T}(\mathbf{Y})$.*

Since the topological product of the spaces $\mathscr{T}(\mathbf{X}_i)$ is not necessarily sequential, continuous realizability only implies sequential continuity.

Admissible representations have the following quotient properties.

Proposition 9.3.9. *Let δ be an admissible representation of a topological space X.*

1) *δ lifts convergent sequences, i.e., for every sequence $(x_n)_n$ converging in X to some $x \in \mathsf{X}$ there is a sequence $(p_n)_n$ converging in $\mathbb{N}^{\mathbb{N}}$ to some $p \in \mathbb{N}^{\mathbb{N}}$ such that $\delta(p_m) = x_m$ for all $m \in \mathbb{N}$ and $\delta(p) = x$.*
2) *δ is a quotient representation of X if, and only if, X is a sequential space.*
3) *δ is an admissible quotient representation of the sequential coreflection of X.*

Although the decimal representation ρ_{dec} is a quotient representation of the Euclidean space $\mathbb{R}$, it does not lift convergent sequences. Hence it is not admissible for $\mathbb{R}$. By contrast, the Cauchy representation enjoys admissibility. This explains why the latter induces a useful notion of computability, whereas the former fails to do so.

Example 9.3.10. The Cauchy representation and the signed-digit representation are admissible for the Euclidean space $\mathbb{R}$, whereas the decimal representation is not. □

Proofs of this subsection can be found in [33].

9.3.5 Examples of Admissible Representations

We present admissible representations of some relevant spaces.

Example 9.3.11 (Admissible representations of relevant metric spaces).
1) The identity on $\mathbb{N}^{\mathbb{N}}$ is an admissible representation of $\mathbb{N}^{\mathbb{N}}$.
2) For the discrete natural numbers $\mathbb{N}$ we use as canonical admissible representation $\rho_{\mathbb{N}}$ the one defined by $\rho_{\mathbb{N}}(p) := p(0)$.
3) The one-point compactification $\mathbb{N}_\infty$ has as carrier set $\mathbb{N} \cup \{\infty\}$. Its topology has as basis the sets $\{a\}$ and $\{\infty, n \mid n \geq a\}$ for all $a \in \mathbb{N}$. An admissible representation $\rho_{\mathbb{N}_\infty}$ for $\mathbb{N}_\infty$ is defined by

$$\rho_{\mathbb{N}_\infty}(p) := \begin{cases} \infty & \text{if } p = 0^\omega \\ \min\{n \in \mathbb{N} \mid p(n) \neq 0\} & \text{otherwise} \end{cases}$$

   for all $p \in \mathbb{N}^{\mathbb{N}}$.
4) Any Polish space X has a total admissible representation. We briefly sketch a construction. Let $d\colon \mathsf{X} \times \mathsf{X} \to \mathbb{R}$ be a complete metric inducing the topology of X, and let $\{a_i \mid i \in \mathbb{N}\}$ be a dense subset of X. Using the continuous function $\ell\colon \mathbb{N}^{\mathbb{N}} \to \mathbb{N}_\infty$ given by

$$\ell(p) := \min\big(\{\infty\} \cup \{m \in \mathbb{N} \mid d(a_{p(m)}, a_{p(m+1)}) \geq 1/2^{m+1}\}\big),$$

   we define a total representation $\delta\colon \mathbb{N}^{\mathbb{N}} \to \mathsf{X}$ by

$$\delta(p) := \begin{cases} \lim\limits_{i\to\infty} a_{p(i)} & \text{if } d(a_{p(m)}, a_{p(m+1)}) < 1/2^{m+1} \text{ for all } m \in \mathbb{N} \\ a_{p(\ell(p))} & \text{otherwise.} \end{cases}$$

   Since $d(\delta(p), a_{p(m)}) \leq 2^{-m}$ holds for all $p \in \mathbb{N}^{\mathbb{N}}$ and $m \leq \ell(p)$, δ is continuous. The maximality property follows from the fact that the restriction of δ to $\ell^{-1}\{\infty\}$ is a version of the Cauchy representation of X similar to the admissible representation defined in [44, Definition 3.4.17]. □

We turn our attention to non-Hausdorff examples.

Example 9.3.12 (Admissible representations of relevant non-Hausdorff spaces).
1) The Sierpiński space $\mathbb{S}$ has $\{\bot, \top\}$ as its carrier set and is topologised by the family $\{\emptyset, \{\top\}, \{\bot, \top\}\}$. Note that the constant sequence $(\top)_n$ converges both to $\top$ and to $\bot$. A total admissible representation $\rho_{\mathbb{S}}$ for $\mathbb{S}$ is defined by

$$\rho_{\mathbb{S}}(p) := \begin{cases} \bot \text{ if } p = 0^\omega \\ \top \text{ otherwise,} \end{cases}$$

where 0^ω denotes the constant sequence consisting of 0's.

2) Scott's graph model $P\omega$ is a prime example of a Scott domain. Its carrier set is the power set of $\mathbb{N}$ ordered by set inclusion $\subseteq$. A countable basis of the Scott topology on $P\omega$ is given by the family of sets $\{M \subseteq \mathbb{N} \,|\, E \subseteq M\}$, where E varies over the finite subsets of $\mathbb{N}$. The enumeration representation $En\colon \mathbb{N}^{\mathbb{N}} \to P\omega$ defined by
$$En(p) := \{p(i) - 1 \,|\, i \in \mathbb{N} \,\&\, p(i) > 0\}$$
is an admissible representation of $P\omega$ viewed as a topological space endowed with the Scott topology (cf. [44]).
3) The *lower reals* $\mathbb{R}_<$ are endowed with the topology $\{(a;\infty) \,|\, a \in \mathbb{R}\} \cup \{\emptyset, \mathbb{R}\}$. An admissible representation $\delta_<$ for $\mathbb{R}_<$ is given by: $\delta_<(p) = x :\iff En(p) = \{i \,|\, \nu_{\mathbb{Q}}(i) < x\}$, where $\nu_{\mathbb{Q}}$ is a canonical numbering of the rational numbers.
4) Any second-countable T_0-space X has an admissible representation (cf. [26]). Given a countable subbase $\{\beta_i \,|\, i \in \mathbb{N}\}$ with $\bigcup_{i\in\mathbb{N}} \beta_i = X$, one defines $\delta_\beta\colon \mathbb{N}^{\mathbb{N}} \dashrightarrow X$ by
$$\delta_\beta(p) = x \;:\iff\; En(p) = \{i \in \mathbb{N} \,|\, x \in \beta_i\}\,.$$
So p is a name of x if, and only if, p lists all (indices of) subbasic sets which contain x. Remember that T_0-spaces are exactly those topological spaces Y such that $\eta_{\mathsf{Y}}\colon y \mapsto \{V \text{ open in } \mathsf{Y} \,|\, y \in V\}$ is injective, ensuring that δ_β is indeed well defined. C. Kreitz and K. Weihrauch called δ_β a *standard representation* for X in [26] and showed that δ_β is continuous and has the maximality property. Hence δ_β is admissible in the sense of Definition 9.3.4.
5) M. de Brecht has characterised the second-countable T_0-spaces enjoying a total admissible representation as the quasi-Polish spaces (see [13]). Quasi-Polishness is a completeness notion for quasi-metric spaces. For example, Scott's graph model $P\omega$ is quasi-Polish. □

9.3.6 The Admissibility Notion of Kreitz and Weihrauch

C. Kreitz and K. Weihrauch introduced a notion of admissible representations tailor-made for representing T_0-spaces X with a countable base. They call a representation ϕ of X *admissible* if ϕ is topologically equivalent to the standard representation δ_β presented in Example 9.3.12. This notion of admissibility guarantees equivalence of continuous realizability and topological continuity for partial multivariate functions between second-countable T_0-spaces. This equivalence is referred to as the Main Theorem in [26, 43, 44]. Moreover, C. Kreitz and K. Weihrauch proved in [26] that ϕ is topologically equivalent to δ_β if, and only if, ϕ is continuous and has the maximality property. Therefore this admissibility notion agrees with the one in Definition 9.3.4 for representations of second-countable T_0-spaces.

9.3.7 Constructing Admissible Representations

In Subsection 9.2.12 we have seen how to form products, coproducts, meets and function spaces in Rep. The corresponding constructions preserve admissibility.

Proposition 9.3.13. *Let δ_i be an admissible representation of a topological space X_i.*

1) *$\delta_1 \boxtimes \delta_2$ is an admissible representation of the topological product of X_1 and X_2.*
2) *$\boxtimes_{i\in\mathbb{N}} \delta_i$ is an admissible representation of the Tychonoff product $\prod_{i\in\mathbb{N}} \mathsf{X}_i$.*
3) *$\delta_1 \boxplus \delta_2$ is an admissible representation of the binary coproduct $\mathsf{X}_1 \uplus \mathsf{X}_2$.*
4) *$\boxplus_{i\in\mathbb{N}} \delta_i$ is an admissible representation of the countable coproduct $\biguplus_{i\in\mathbb{N}} \mathsf{X}_i$.*
5) *$\delta_1 \wedge \delta_2$ is an admissible representation of the meet $(X_1 \cap X_2, O(\mathsf{X}_1) \wedge O(\mathsf{X}_2))$.*
6) *$\wedge_{i\in\mathbb{N}} \delta_i$ is an admissible representation of the meet $(\bigcap_{i\in\mathbb{N}} \mathsf{X}_i, \bigwedge_{i\in\mathbb{N}} O(\mathsf{X}_i))$.*

Here $\bigwedge_{i\in\mathbb{N}} O(\mathsf{X}_i)$ denotes the countable meet topology which is generated by the basis

$$\left\{ \bigcap_{i=0}^{k} U_i \,\middle|\, k \in \mathbb{N},\ U_i \text{ open in } \mathsf{X}_i \text{ for all } i \leq k \right\}.$$

The binary meet topology $O(\mathsf{X}_1) \wedge O(\mathsf{X}_2)$ is defined analogously.

For the function space representation to be admissible it suffices that the target space representation is admissible, whereas the source space representation only has to have appropriate quotient properties. This fact is crucial in showing that every qcb_0-space has an admissible representation (cf. Subsection 9.4.4). Recall that any admissible representation lifts convergent sequences; moreover, it is a quotient representation whenever the represented topological space is sequential (cf. Proposition 9.3.9).

Proposition 9.3.14. *Let γ be an admissible representation of a topological space Y.*

1) *Let δ be a quotient representation of a topological space X. Then $[\delta \to \gamma]$ is an admissible representation of the space of all total topologically continuous functions from X to Y endowed with the compact-open topology.*
2) *Let δ be a continuous representation of a topological space X that lifts convergent sequences. Then $[\delta \to \gamma]$ is an admissible representation of the space of all total sequentially continuous functions from X to Y endowed with the compact-open topology.*

Now we turn our attention to subspaces and sequential embeddings. A function $e\colon \mathsf{Z} \to \mathsf{X}$ is called a *sequential embedding* of Z into X if e is injective and satisfies $(z_n)_n \to_{\mathsf{Z}} z \iff (e(z_n))_n \to_{\mathsf{X}} e(z)$.

Proposition 9.3.15. *Let δ be an admissible representation of a topological space X.*

1) *Let Y be a topological subspace of X. Then the corestriction $\delta|^{\mathsf{Y}}$ is an admissible representation of Y.*
2) *Let $e\colon \mathsf{Z} \to \mathsf{X}$ be a sequential embedding of a topological space Z into X. Then $\zeta\colon \mathbb{N}^{\mathbb{N}} \dashrightarrow \mathsf{Z}$ defined by $\zeta(p) = z :\iff e(z) = \delta(p)$ is an admissible representation of Z.*

For proofs we refer to [33, 35].

9.3.8 Pseudobases

The class of topological spaces enjoying an admissible representation can be characterised with the help of the notion of a pseudobase. A family $\mathscr{B}$ of subsets of a topological space X is called a *pseudobase* for X if for every open set U in X and for every sequence $(x_n)_n$ converging to some element $x \in U$ there is a set $B \in \mathscr{B}$ with

$$x \in B \subseteq U \quad \text{and} \quad x_n \in B \quad \text{for almost all } n \geq n_0.$$

A family $\mathscr{B}$ of subsets such that its closure under finite intersection is a pseudobase for X is called a *pseudosubbase* for X. Any base of a topological space is a pseudobase; the converse is not true in general. Pseudobases become interesting when they are countable. Given pseudobases for spaces X and Y, there exist canonical constructions for forming pseudobases for product, coproduct, meet, exponentiation and subspaces (cf. [18, 33, 35]).

Example 9.3.16 (Pseudobase generated by an admissible representation).
Let δ be an admissible representation of a topological space X. Then the sets $\delta[u\mathbb{N}^{\mathbb{N}} \cap \mathrm{dom}(\delta)]$, where u varies over the finite strings over $\mathbb{N}$, form a countable pseudobase for X. □

Together with the T_0-property, the existence of a countable pseudobase characterises the class of admissibly representable topological spaces.

Proposition 9.3.17 (Topological spaces with admissible representations).
1) *A topological space has an admissible representation if, and only if, it has a countable pseudobase and satisfies the T_0-property.*
2) *A topological space has an admissible quotient representation if, and only if, it is sequential, has a countable pseudobase and satisfies the T_0-property.*

Given a countable pseudobase $\{\beta_i \mid i \in \mathbb{N}\}$ for a T_0-space X, an admissible representation θ_β for X can be constructed by

$$\theta_\beta(p) = x \;:\Longleftrightarrow\; \begin{cases} En(p) \subseteq \{i \in \mathbb{N} \mid x \in \beta_i\} & \& \\ \forall U \in \eta_{\mathsf{X}}(x).\exists i \in En(p).\beta_i \subseteq U \end{cases}$$

where $\eta_{\mathsf{X}}(x)$ denotes the family of open sets containing x and En is the enumeration numbering of $\{M \mid M \subseteq \mathbb{N}\}$ from Example 9.3.12. The T_0-property guarantees that at most one element x satisfies the right-hand side of the displayed formula. On the other hand, a simple cardinality argument shows that the existence of an admissible representation implies the T_0-property.

9.4 Qcb-Spaces

We discuss the class of qcb-spaces and their subclass of qcb_0-spaces. Qcb_0-spaces are shown to be exactly the class of topological spaces that have an admissible quotient representation. Their excellent closure properties identify them as an appropriate topological framework for studying computability on spaces with cardinality of the continuum. Moreover, we investigate computability properties of qcb-spaces using the notion of an effective qcb-space. Finally, we present quasi-normal spaces as an important subclass of Hausdorff qcb-spaces suitable for studying computability in Functional Analysis.

9.4.1 Qcb-Spaces

A topological space X is called a *qcb-space* (*q*uotient of a *c*ountably *b*ased space) if X can be exhibited as a topological quotient of a topological space W with a countable base. This means that there exists a surjection $q\colon \mathsf{W} \twoheadrightarrow \mathsf{X}$ such that X carries the finest (= largest) topology τ_q such that q is topologically continuous (in this case q is called a *quotient map*). Clearly, this finest topology is equal to $\{U \subseteq \mathsf{X} \,|\, q^{-1}[U] \text{ open in } \mathsf{W}\}$. Because it is a topological quotient of a sequential space, any qcb-space is sequential as well (cf. [15]). Recall that this implies that a total unary function between qcb-spaces is topologically continuous if, and only if, it is sequentially continuous. We write QCB for the category of qcb-spaces with continuous functions as morphisms and QCB_0 for the full subcategory of *qcb*$_0$*-spaces*, which are qcb-spaces satisfying the T_0-property.

Clearly, any separable metrisable space is a qcb_0-space because it is second-countable and Hausdorff. Moreover, the topological space associated with a represented space is a qcb-space. So henceforth we consider the functor $\mathscr{T}$ from Subsection 9.3.1 as a functor from Rep to QCB.

The category QCB was first investigated in [33] under the acronym AdmSeq. In [3], QCB was shown to be the same category as the category PQ of sequential spaces with an ω-projecting cover. Later the name 'qcb-space' was coined by A. Simpson in [40]. Further information about QCB can be found in [1, 18, 29].

9.4.2 Operators on Qcb-Spaces

The category QCB of qcb-spaces and its full subcategory QCB_0 have excellent closure properties.

Theorem 9.4.1. *The categories* QCB *and* QCB_0 *are cartesian closed and have all countable limits and all countable colimits.*

For short this means that we can form within QCB and QCB_0 countable products, function spaces, subspaces, countable coproducts and quotients.

We sketch how to construct qcb-spaces from others. Unless stated otherwise, the respective operators preserve the T_0-property, hence they apply both to QCB and QCB_0.

Example 9.4.2 (Construction of qcb-spaces).

1) *Binary products:* The QCB-product of X and Y is the sequential coreflection (see Subsection 9.3.3) of the topological product $\mathsf{X} \times_{\mathsf{Top}} \mathsf{Y}$.
2) *Countable products:* Similarly, the QCB-product $\prod_{i\in\mathbb{N}} \mathsf{X}_i$ of a sequence $(\mathsf{X}_i)_{i\in\mathbb{N}}$ of qcb-spaces is the sequential coreflection of the Tychonoff product of these spaces. The countable product $\prod_{i\in\mathbb{N}} \mathsf{X}$ of a single qcb-space X with itself is homeomorphic to the QCB-exponential $\mathsf{X}^{\mathbb{N}}$ (see below), where $\mathbb{N}$ are the natural numbers equipped with the discrete topology.
3) *Exponentiation:* Given two qcb-spaces X and Y, the carrier set of the QCB-exponential, which we denote by Y^{X}, is the set $C(\mathsf{X},\mathsf{Y})$ of all total continuous functions $f\colon \mathsf{X} \to \mathsf{Y}$. The topology is given by the sequentialisation of the compact-open topology on $C(\mathsf{X},\mathsf{Y})$. The compact-open topology is generated by the subbasic opens

$$\langle K, V\rangle := \left\{ f \in C(\mathsf{X},\mathsf{Y}) \,\middle|\, f[K] \subseteq V \right\},$$

where K varies over the compact subsets of X and V over the open subsets of Y. For Y^{X} to be a T_0-space it suffices that Y has the T_0-property. The convergence relation on Y^{X} is continuous convergence: $(f_n)_n$ *converges continuously to* f if $(f_n(x_n))_n$ converges in Y to $f(x)$, whenever $(x_n)_n$ is a sequence converging in X to $x \in \mathsf{X}$.
4) *Subspaces:* Given a qcb-space X and a subset Z of X, the subspace topology on Z inherited from X is not necessarily sequential. Instead one has to consider the *subsequential topology* on Z, which is defined as the sequentialisation of the subspace topology. This topology is indeed a qcb-topology.
5) *Equalisers:* Let $f, g\colon \mathsf{X} \to \mathsf{Y}$ be continuous functions between $\mathsf{X},\mathsf{Y} \in \mathsf{QCB}$. The set $E := \{x \in \mathsf{X} \,|\, f(x) = g(x)\}$ endowed with the subsequential topology along with the continuous inclusion map forms an equaliser to f, g in QCB.
6) *Countable limits:* Since QCB and QCB_0 have countable products and equalisers, both have all countable limits. Hence both categories are countably complete.
7) *Meets:* Given a sequence of qcb-spaces $(\mathsf{X}_i)_i$, their *meet* (or *conjunction*) $\bigwedge_{i\in\mathbb{N}} \mathsf{X}_i$ is defined to have the intersection as its carrier set. The topology is the sequentialisation of the meet topology which is generated by the basis

$$\left\{ \bigcap_{i=0}^{k} U_i \,\middle|\, k \in \mathbb{N},\ U_i \text{ open in } \mathsf{X}_i \text{ for all } i \le k \right\}.$$

Since this is a countable limit construction, $\bigwedge_{i\in\mathbb{N}} \mathsf{X}_i$ is a qcb-space. The binary meet $\mathsf{X}_1 \wedge \mathsf{X}_2$ is defined analogously. Note that the Euclidean reals $\mathbb{R}$ arises as the meet of the lower reals $\mathbb{R}_<$ and the upper reals $\mathbb{R}_>$.

8) *Countable coproduct:* The countable coproduct of a sequence $(\mathsf{X}_i)_{i\in\mathbb{N}}$ of qcb-spaces is constructed as in the category Top of topological spaces. The carrier set is the disjoint union $\bigcup_{i\in\mathbb{N}}\{(i,x)\,|\,x\in\mathsf{X}_i\}$. A subset U is open if, and only if, for every $i\in\mathbb{N}$ the set $\{x\in\mathsf{X}_i\,|\,(i,x)\in U\}$ is open in X_i.
9) *Quotients:* For any qcb-space X and any quotient map $q:\mathsf{X}\twoheadrightarrow\mathsf{Z}$ the topological space Z is a qcb-space as well. But in general Z does not have the T_0-property, even if X is T_0.
10) *Kolmogorov quotient ("T_0-fication"):* For a qcb-space X the equivalence relation of topological indistinguishability is defined by $x\equiv_{\mathsf{X}} x' :\Longleftrightarrow \eta_{\mathsf{X}}(x)=\eta_{\mathsf{X}}(x')$, where $\eta_{\mathsf{X}}(x)$ denotes the neighbourhood filter $\{U \text{ open}\,|\,x\in U\}$ of x. The Kolmogorov quotient of a T_0-space is just the space itself. If X is not T_0, then the *Kolmogorov quotient* $\mathsf{KQ}(\mathsf{X})$ of X consists of the equivalence classes of $\equiv_{\mathsf{X}}$ and its topology is the quotient topology induced by the surjection $[.]_{\equiv_{\mathsf{X}}}$ that maps x to its equivalence class.
11) *Coequalisers in* QCB*:* Let $f,g\colon\mathsf{A}\to\mathsf{X}$ be continuous functions between $\mathsf{A},\mathsf{X}\in\mathsf{QCB}$. We consider the transitive hull $\approx$ of the reflexive relation

$$\big\{(x,x)\,\big|\,x\in\mathsf{X}\big\}\cup\big\{(f(a),g(a))\,\big|\,a\in\mathsf{A}\big\}\cup\big\{(g(a),f(a))\,\big|\,a\in\mathsf{A}\big\},$$

define Z as the space of equivalence classes of $\approx$ and endow Z with the quotient topology induced by the surjection $[.]_{\approx}$ that maps x to its equivalence class. Then Z and $[.]_{\approx}$ form a coequaliser to f,g in QCB.
12) *Coequalisers in* QCB_0*:* Given two continuous functions $f,g\colon\mathsf{A}\to\mathsf{X}$ between spaces $\mathsf{A},\mathsf{X}\in\mathsf{QCB}_0$, we first construct Z and $[.]_{\approx}$ as above. If Z is a T_0-space, then this pair is also a coequaliser in QCB_0. Otherwise we take the Kolmogorov quotient of Z together with the composition $[.]_{\equiv_{\mathsf{Z}}}\circ[.]_{\approx}$.
13) *Countable colimits:* Since QCB and QCB_0 have countable coproducts and coequalisers, both have all countable colimits. Hence both categories are countably cocomplete. □

More information can be found in [1, 18, 33].

9.4.3 Powerspaces in QCB_0

We discuss spaces of families of subsets of qcb-spaces, including open, closed, overt and compact subsets.

9.4.3.1 Open subsets

Given a qcb-space X, we endow the family $\mathcal{O}(\mathsf{X})$ of open sets of X with the Scott-topology on the complete lattice $(\mathcal{O}(\mathsf{X}),\subseteq)$ and denote the resulting topological space by $\mathscr{O}(\mathsf{X})$. A subset $H\subseteq O(X)$ is called *Scott open* if H is upwards closed and $D\cap H$ is non-empty for each directed subset $D\subseteq O(\mathsf{X})$ with $\bigcup(D)\in H$. Since any

qcb-space is hereditarily Lindelöf, the Scott topology coincides with the ω-Scott topology. The *ω-Scott topology* consists of all upwards-closed subsets H such that for all sequences $(U_i)_{i\in\mathbb{N}}$ with $\bigcup_{i\in\mathbb{N}} U_i \in H$ there exists some n such that $\bigcup_{i=0}^{n} U_i \in H$.

The space $\mathscr{O}(\mathsf{X})$ is homeomorphic to the QCB-exponential $\mathbb{S}^{\mathsf{X}}$, where $\mathbb{S}$ denotes the Sierpiński space (cf. Example 9.3.12). The homeomorphism is given by the natural identification of an open set U with its continuous characteristic function $cf(U)\colon \mathsf{X}\to\mathbb{S}$ defined by $cf(U)(x)=\top :\Longleftrightarrow x\in U$.

Proposition 9.4.3. *Let X be a qcb-space.*

1) *The space $\mathscr{O}(\mathsf{X})$ is a qcb$_0$-space.*
2) *The space $\mathscr{O}(\mathsf{X})$ is homeomorphic to $\mathbb{S}^{\mathsf{X}}$ via the map $cf\colon O(\mathsf{X})\to\mathbb{S}^{\mathsf{X}}$.*
3) *The Scott topology on $O(\mathsf{X})$ is the sequentialisation of the* compact-open *topology on $O(\mathsf{X})$, which is generated by basic sets $K^{\subseteq} := \{U\in O(\mathsf{X}) \mid K\subseteq U\}$, where K varies over the compact subsets of X.*
4) *A sequence $(U_n)_n$ converges to V in $\mathscr{O}(\mathsf{X})$ if, and only if,*

$$W_k := \bigcap_{n\geq k} U_n \cap V \text{ is open for all } k\in\mathbb{N} \text{ and } V = \bigcup_{n\in\mathbb{N}} W_n.$$

5) *Binary intersection $\cap\colon \mathscr{O}(\mathsf{X})\times\mathscr{O}(\mathsf{X})\to\mathscr{O}(\mathsf{X})$ is sequentially continuous.*
6) *Countable union $\bigcup\colon \prod_{i\in\mathbb{N}}\mathscr{O}(\mathsf{X})\to\mathscr{O}(\mathsf{X})$ is sequentially continuous.*

A space such that the Scott topology on its lattice of open subsets agrees with the compact-open topology is called *consonant*. M. de Brecht has shown that quasi-Polish spaces are consonant (cf. [14]), whereas the space of rational numbers is known to be dissonant (= non-consonant).

Importantly, any qcb$_0$-space X embeds sequentially into the space $\mathscr{O}(\mathscr{O}(\mathsf{X}))$ via the neighbourhood filter map η_{X}. Recall that this means that η_{X} is injective and $(x_n)_n$ converges to x in X if, and only if, $(\eta_{\mathsf{X}}(x_n))_n$ converges to $\eta_{\mathsf{X}}(x)$ in $\mathscr{O}(\mathscr{O}(\mathsf{X}))$.

Proposition 9.4.4. *Let X be a qcb$_0$-space.*

1) *For every $x\in\mathsf{X}$, the open neighbourhood filter $\eta_{\mathsf{X}}(x) := \{U \text{ open} \mid x\in U\}$ is open in $\mathscr{O}(\mathsf{X})$.*
2) *The map $\eta_{\mathsf{X}}\colon \mathsf{X}\to\mathscr{O}(\mathscr{O}(\mathsf{X}))$ is a sequential embedding of X into $\mathscr{O}(\mathscr{O}(\mathsf{X}))$.*
3) *The map $\eta_{\mathsf{X}}\colon \mathsf{X}\to\mathbb{S}^{\mathbb{S}^{\mathsf{X}}}$ defined by $\eta_{\mathsf{X}}(x)(h) := h(x)$ is a sequential embedding of X into the QCB-exponential $\mathbb{S}^{\mathbb{S}^{\mathsf{X}}}$.*

Recall that $\mathscr{O}(\mathscr{O}(\mathsf{X}))$ is homeomorphic to $\mathbb{S}^{\mathbb{S}^{\mathsf{X}}}$. We remark that η_{X} is even a topological embedding of X into $\mathbb{S}^{\mathbb{S}^{\mathsf{X}}}$, meaning that η_{X} forms a homeomorphism between X and its image endowed with the subspace topology. For more information see [14, 36].

9.4.3.2 Overt subsets

On the family of all subsets of X we consider the topology of *overtness*, which has as subbasis all sets of the form $\Diamond U := \{M \subseteq \mathsf{X} \,|\, M \cap U \neq \emptyset\}$, where U varies over the open sets of X. By $\mathscr{V}(\mathsf{X})$ we denote the space all subsets of X equipped with the sequentialisation of this topology.

Proposition 9.4.5. *Let X be a qcb-space.*

1) *$\mathscr{V}(\mathsf{X})$ is a qcb-space.*
2) *For every subset M, $\eta^{\Diamond}_{\mathsf{X}}(M) := \{U \text{ open} \,|\, M \in \Diamond U\}$ is open in $\mathscr{O}(\mathsf{X})$.*
3) *Binary union on $\mathscr{V}(\mathsf{X})$ is sequentially continuous.*
4) *Binary intersection on $\mathscr{V}(\mathbb{R})$ is sequentially discontinuous.*
5) *Two subsets $M_1, M_2 \subseteq \mathsf{X}$ have the same neighbourhoods in $\mathscr{V}(\mathsf{X})$ if, and only if, their topological closure agrees.*

Overt subspaces were investigated by P. Taylor in [41].

9.4.3.3 Closed subsets

The upper Fell topology on the family of closed subsets of a qcb-space X is generated by the basic sets $\{A \text{ closed} \,|\, A \cap K = \emptyset\}$, where K varies over the compact subsets of X. This topology is not sequential, unless X is a consonant space. By $\mathscr{A}_{-}(\mathsf{X})$ we denote the space of closed subspaces equipped with the sequentialisation of the upper Fell topology. Clearly $\mathscr{A}_{-}(\mathsf{X})$ is homeomorphic to $\mathscr{O}(\mathsf{X})$ via set complementation. So $\mathscr{A}_{-}(\mathsf{X})$ is a qcb_0-space.

The lower Fell topology on $A(\mathsf{X})$ has as subbasis all sets of the form $\Diamond U := \{A \text{ closed} \,|\, A \cap U \neq \emptyset\}$. By $\mathscr{A}_{+}(\mathsf{X})$ we denote the space of closed sets of X carrying the sequentialisation of the lower Fell topology on X. This space is homeomorphic to the Kolmogorov quotient of the space $\mathscr{V}(\mathsf{X})$ from Subsection 9.4.3.2.

Proposition 9.4.6. *Let X be a qcb-space.*

1) *$\mathscr{A}_{+}(\mathsf{X})$ is a qcb_0-space.*
2) *The function $\eta^{\Diamond}_{\mathsf{X}} \colon \mathscr{A}_{+}(\mathsf{X}) \mapsto \mathscr{O}(\mathscr{O}(\mathsf{X}))$ mapping A to $\{U \text{ open} \,|\, A \in \Diamond U\}$ is a sequential embedding of $\mathscr{A}_{+}(\mathsf{X})$ into $\mathscr{O}(\mathscr{O}(\mathsf{X}))$.*
3) *Binary union on $\mathscr{A}_{+}(\mathsf{X})$ is sequentially continuous.*
4) *Binary intersection on $\mathscr{A}_{+}(\mathbb{R})$ is sequentially discontinuous.*

Computability on closed subsets of computable metric spaces was studied by V. Brattka and G. Presser in [12]. P. Hertling ([24]) and M. Ziegler ([49]) investigated computability of regular closed subsets of $\mathbb{R}^d$.

9.4.3.4 Compact subsets

Several notions of compactness are studied in topology: a subset K of a topological space is *compact* if every open cover of K has a finite subcover; K is called *countably compact* if every countable cover of K has a finite subcover; K is called *sequentially compact* if every sequence $(x_n)_n$ in K has a subsequence that converges to some element in K. For subsets of qcb-spaces, however, these three notions coincide (cf. [2]).

The *Vietoris topology* on the family of all compact subsets of a topological space X is generated by the subbasic open sets

$$\Box U := \{K \text{ compact in } \mathsf{X} \,|\, K \subseteq U\} \text{ and } \Diamond U := \{K \text{ compact in } \mathsf{X} \,|\, K \cap U \neq \emptyset\},$$

where U varies over the open subsets of X. By $\mathscr{K}_{\mathrm{Viet}}(\mathsf{X})$ we denote the space of all compact subsets of X equipped with the sequentialisation of the Vietoris topology.

The *upper Vietoris topology* on the family of compact sets is generated by the basic open sets $\Box U$ only. Unless X is a T_1-space, the upper Vietoris topology does not enjoy the T_0-property. Therefore we restrict ourselves to the family of all saturated compact subsets of X, on which the corresponding upper Vietoris topology is indeed T_0. A subset M is *saturated* if, and only if, it is equal to its *saturation* $\uparrow M$, which is the intersection of all open sets containing M. In a T_1-space the saturation of a set yields the set itself. By $\mathscr{K}(X)$ we denote the space of compact saturated subsets of X equipped with the sequentialisation of the upper Vietoris topology.

Proposition 9.4.7. *Let X be a qcb$_0$-space.*

1) *$\mathscr{K}(\mathsf{X})$ and $\mathscr{K}_{\mathrm{Viet}}(\mathsf{X})$ are qcb$_0$-spaces.*
2) *For every compact subset K, $\eta_{\mathsf{X}}^{\Box}(K) := \{U \text{ open} \,|\, K \in \Box U\}$ is open in $\mathscr{O}(\mathsf{X})$.*
3) *The map $\eta_{\mathsf{X}}^{\Box}$ yields a sequential embedding of $\mathscr{K}(\mathsf{X})$ into $\mathscr{O}(\mathscr{O}(\mathsf{X}))$.*
4) *Binary union on $\mathscr{K}(\mathsf{X})$ and on $\mathscr{K}_{\mathrm{Viet}}(\mathsf{X})$ are sequentially continuous.*

Note that the intersection of two saturated compact sets need not be compact, unless X is for example Hausdorff.

The idea to compute on compact subsets via their embedding into $\mathscr{O}(\mathscr{O}(\mathsf{X}))$ was independently developed by M. Escardó in [16, 17] and M. Schröder in [33]. V. Brattka and G. Presser investigated computability on compact subsets of computable metric spaces in [12].

9.4.4 Qcb-Spaces and Admissibility

Qcb$_0$-spaces form exactly the class of sequential spaces that have an admissible representation. For the proof, one first observes that any qcb$_0$-space has a quotient representation: if a T_0-space X is a topological quotient space of a second-countable space W, then X is also a topological quotient of the Kolmogorov quotient $\mathsf{KQ}(\mathsf{W})$

of W. By Example 9.3.12, $\mathsf{KQ}(\mathsf{W})$ has an admissible representation; its composition with the quotient map $\mathsf{KQ}(\mathsf{W}) \twoheadrightarrow \mathsf{X}$ yields a quotient representation of X.

Now we define an operator that converts a quotient representation δ into an admissible representation. From Propositions 9.3.14, 9.3.15 and 9.4.4 we know for a quotient representation δ of a T_0-space X:

a) X embeds sequentially into the QCB_0-exponential $\mathbb{S}^{\mathbb{S}^{\mathsf{X}}}$ via η_{X}.
b) $[\delta \to \rho_{\mathbb{S}}]$ is an admissible representation of $\mathbb{S}^{\mathsf{X}}$, even if δ is not admissible.
c) Corestrictions of admissible representations are admissible again.

These observations motivate us to define an operator by

$$\delta^{\mathbb{S}}(p) = x \;:\Longleftrightarrow\; \eta_{\mathsf{X}}(x) = [[\delta \to \rho_{\mathbb{S}}] \to \rho_{\mathbb{S}}](p)\,.$$

The above facts imply that $\delta^{\mathbb{S}}$ is admissible. In the case that the final topology τ_δ of δ lacks the T_0-property, it makes sense to generalise this definition by letting $\delta^{\mathbb{S}}$ be the representation of the Kolmogorov quotient $\mathsf{KQ}(X, \tau_\delta)$ given by the equivalence

$$\delta^{\mathbb{S}}(p) = [x]_{\equiv} \;:\Longleftrightarrow\; \eta_{(X,\tau_\delta)}(x) = [[\delta \to \rho_{\mathbb{S}}] \to \rho_{\mathbb{S}}](p)\,,$$

where $\equiv$ denotes the equivalence relation of topological indistinguishability (see Example 9.4.2). In any case, the map $\delta^{\mathbb{S}}$ is indeed an admissible representation.

Theorem 9.4.8 (Admissibility of $\delta^{\mathbb{S}}$).
Let δ be a representation of a set X.

1) *$\delta^{\mathbb{S}}$ is an admissible representation of (X, τ_δ) if τ_δ has the T_0-property.*
2) *$\delta^{\mathbb{S}}$ is an admissible representation of $\mathsf{KQ}(X, \tau_\delta)$.*
3) *If δ is a quotient representation of a T_0-space X, then $\delta^{\mathbb{S}}$ is an admissible representation of X.*

We obtain the following characterisation of admissible representations.

Theorem 9.4.9 (Characterisation of admissibility). *A representation δ of a set X is an admissible representation of (X, τ_δ) if, and only if, δ is topologically equivalent to $\delta^{\mathbb{S}}$ and τ_δ satisfies the T_0-property.*

Along with Propositions 9.3.9 and 9.3.17 we obtain the following characterisation.

Theorem 9.4.10 (Sequential spaces with admissible representation).

1) *A sequential topological space has an admissible representation if, and only if, it is a qcb_0-space.*
2) *A topological space has an admissible quotient representation if, and only if, it is a qcb_0-space.*
3) *A topological space has an admissible representation if, and only if, its sequential coreflection is a qcb_0-space.*

From our previous characterisation in Proposition 9.3.17 we deduce that a sequential space is a qcb-space if, and only if, it has a countable pseudobase. We remark that there are canonical ways to construct pseudobases for qcb-spaces that are built by the operators considered in Example 9.4.2 and Subsection 9.4.3 (cf. [33, 35]). Theorem 9.4.10 implies that the category of admissibly represented spaces and continuously realizable functions is equivalent to QCB_0 (cf. [2]). Proofs of this subsection can be found in [33].

9.4.5 Effectively Admissible Representations

The basic observation in Theorem 9.4.9 that a representation δ is topologically admissible if, and only if, it is topologically equivalent to $\delta^{\mathbb{S}}$ leads one to call a representation δ *effectively admissible* if δ is even computably equivalent to $\delta^{\mathbb{S}}$.

Proposition 9.4.11. *Let δ,γ be representations such that their final topologies are T_0.*

1) *Any (δ,γ)-computable total function f is $(\delta^{\mathbb{S}},\gamma^{\mathbb{S}})$-computable.*
2) *Any (δ,γ)-continuous total function f is $(\delta^{\mathbb{S}},\gamma^{\mathbb{S}})$-continuous.*
3) *The representations $\delta^{\mathbb{S}}$ and $\gamma^{\mathbb{S}}$ are effectively admissible.*
4) *The signed-digit representation and the Cauchy representation of $\mathbb{R}$ are effectively admissible.*

We remark that Proposition 9.4.11(1) does not hold for partial functions: it is even possible that a (δ,γ)-computable partial function is not $(\delta^{\mathbb{S}},\gamma^{\mathbb{S}})$-continuous.

Proposition 9.4.12. *Let δ and γ be effectively admissible representations of qcb_0-spaces X and Y, respectively.*

1) *$\delta\boxtimes\gamma$ is an effectively admissible representation of the QCB-product $\mathsf{X}\times\mathsf{Y}$.*
2) *$\delta\boxplus\gamma$ is an effectively admissible representation of the QCB-coproduct $\mathsf{X}\uplus\mathsf{Y}$.*
3) *$[\delta\to\gamma]$ is an effectively admissible representation of the QCB-exponential Y^{X}.*
4) *$\delta\wedge\gamma$ is an effectively admissible representation of the QCB-meet $\mathsf{X}\wedge\mathsf{Y}$.*
5) *Let Z be a subsequential subspace of X.*
   Then the corestriction $\delta|^{\mathsf{Z}}$ is an effectively admissible representation of Z.
6) *Let $e\colon\mathsf{Z}\to\mathsf{X}$ be a sequential embedding of a sequential space Z into X.*
   Then $e^{-1}\circ\delta$ is an effectively admissible representation of Z.

To construct effectively admissible representations for the function space Y^{X} and for the power spaces over X considered in Subsection 9.4.3, we only need a quotient representation of X.

Proposition 9.4.13. *Let δ be a quotient representation of a qcb_0-space X.*

1) *Let γ be an effectively admissible representation of a qcb_0-space Y.*
   Then $[\delta\to\gamma]$ is an effectively admissible representation of Y^{X}.

2) *Let* $q\colon \mathsf{X} \to \mathsf{Z}$ *be a quotient map. Then* $(q \circ \delta)^{\mathbb{S}}$ *is an effectively admissible representation of the Kolmogorov quotient of* Z.
3) *Define* $\delta^{\mathscr{O}}$ *by* $\delta^{\mathscr{O}}(p) = U :\iff cf(U) = [\delta \to \rho_{\mathbb{S}}](p)$.
Then $\delta^{\mathscr{O}}$ *is an effectively admissible representation of* $\mathscr{O}(\mathsf{X})$.
4) *Define* $\delta^{\mathscr{A}_-}$ *by* $\delta^{\mathscr{A}_-}(p) = A :\iff cf(X \setminus A) = [\delta \to \rho_{\mathbb{S}}](p)$.
Then $\delta^{\mathscr{A}_-}$ *is an effectively admissible representation of* $\mathscr{A}_-(\mathsf{X})$.
5) *Define* $\delta^{\mathscr{A}_+}$ *by* $\delta^{\mathscr{A}_+}(p) = A :\iff \eta_{\mathsf{X}}^{\Diamond}(A) = [[\delta \to \rho_{\mathbb{S}}] \to \rho_{\mathbb{S}}](p)$.
Then $\delta^{\mathscr{A}_+}$ *is an effectively admissible representation of* $\mathscr{A}_+(\mathsf{X})$.
6) *Define* $\delta^{\mathscr{K}}$ *by* $\delta^{\mathscr{K}}(p) = K :\iff \eta_{\mathsf{X}}^{\Box}(K) = [[\delta \to \rho_{\mathbb{S}}] \to \rho_{\mathbb{S}}](p)$.
Then $\delta^{\mathscr{K}}$ *is an effectively admissible representation of* $\mathscr{K}(\mathsf{X})$.
7) *Define* $\delta^{\mathscr{K}_{\mathrm{Viet}}}$ *by* $\delta^{\mathscr{K}_{\mathrm{Viet}}}(p) = K :\iff \delta^{\mathscr{K}}(\pi_1(p)) = K \,\&\, \delta^{\mathscr{A}_+}(\pi_2(p)) = \mathrm{Cls}(K)$, *where* $\mathrm{Cls}(K)$ *denotes the topological closure of* K. *Then* $\delta^{\mathscr{K}_{\mathrm{Viet}}}$ *is an effectively admissible representation of* $\mathscr{K}_{\mathrm{Viet}}(\mathsf{X})$.

Proofs of this subsection can be found in [33].

9.4.6 Effective Qcb-Spaces

An *effective qcb-space* is a represented space $\mathbf{X} = (X, \delta)$ such that τ_δ has the T_0-property and δ is an effectively admissible representation of the associated qcb-space (X, τ_δ). The category EffQCB has effective qcb-spaces as its objects and total computable functions as morphisms. So EffQCB is a full subcategory of $\mathsf{Rep}_{\mathrm{eff}}$.

A similar category was first investigated in [33, 34] under the acronym EffSeq, but it comprises also effectively multirepresented qcb-spaces by dropping the T_0-condition. A. Pauly has investigated effective qcb-spaces under the name *computably admissible represented space* (cf. [32]). However, we prefer the former name, because there exist further classes of represented spaces enjoying other reasonable effectivity properties. We will discuss this in Subsections 9.4.9 and 9.5.3.

Example 9.4.14 (Basic topological spaces as effective qcb-spaces).
The following are effective qcb-spaces.

1) $(\mathbb{N}^{\mathbb{N}}, id_{\mathbb{N}^{\mathbb{N}}})$.
2) $(\mathbb{R}, \rho_{\mathbb{R}})$, where $\rho_{\mathbb{R}}$ is the Cauchy representation from Subsection 9.2.7.
3) $(\mathbb{R}, \rho_{\mathrm{sd}})$, where ρ_{sd} is the signed-digit representation from Subsection 9.2.8.
4) $(\mathbb{N}, \rho_{\mathbb{N}})$ with $\rho_{\mathbb{N}}(p) = p(0)$.
5) $(\mathbb{S}, \rho_{\mathbb{S}})$ with $\rho_{\mathbb{S}}(p) = \bot :\iff p = 0^\omega$ and $\rho_{\mathbb{S}}(p) = \top :\iff p \neq 0^\omega$. □

Computable topological spaces in the sense of T. Grubba and K. Weihrauch are effective qcb-spaces.

Example 9.4.15 (Computable topological spaces).
A *computable topological space* (cf. [47]) is a pair (X, β), where β is a partial numbering of a base of a T_0-topology such that there is a computably enumerable set

$J \subseteq \mathbb{N}^3$ satisfying $\beta_a \cap \beta_b = \bigcup\{\beta_k \,|\, (a,b,k) \in J\}$ and the domain of β is computably enumerable. The standard representation δ_β from Example 9.3.12 defined by

$$\delta_\beta(p) = x \;:\Longleftrightarrow\; En(p) = \{i \in \mathrm{dom}(\beta) \,|\, x \in \beta_i\}$$

is effectively admissible. So (X, δ_β) is an effective qcb-space. □

Computable metric spaces can be viewed as computable topological spaces in the sense of Example 9.4.15 and thus as effective qcb-spaces.

Example 9.4.16 (Computable metric space).
A *computable metric space (cms)* is a metric space (X,d) together with a total numbering α of a dense subset A such that the function $(i,j) \mapsto d(\alpha_i, \alpha_j)$ is $(\rho_\mathbb{N} \boxtimes \rho_\mathbb{N}; \rho_\mathbb{R})$-computable. The latter is equivalent to saying that $(d(\alpha_{fst(k)}, \alpha_{snd(k)}))_k$ is a computable sequence of real numbers, where $fst, snd\colon \mathbb{N} \to \mathbb{N}$ denote the computable projections of a canonical computable pairing function for $\mathbb{N}$.

A canonical numbering β of a base of the topology generated by the metric d is constructed by

$$\beta_k := \{x \in X \,|\, d(x, \alpha_{fst(k)}) < 2^{-snd(k)}\},$$

The computability condition on α guarantees that (X,β) is a computable topological space in the sense of Example 9.4.15. The *Cauchy representation* $\rho_\alpha\colon \mathbb{N}^\mathbb{N} \dashrightarrow X$ is defined by

$$\rho_\alpha(p) = x \;:\Longleftrightarrow\; \forall k \in \mathbb{N}.\, d(x, \alpha_{p(k)}) \leq 2^{-k}$$

for all $p \in \mathbb{N}$ and $x \in X$. The Cauchy representation is computably equivalent to the standard representation δ_β from Example 9.4.15. Hence (X, ρ_α) is an effective qcb-space.

The real numbers $\mathbb{R}$ together with the numbering $\nu_\mathbb{Q}$ from Subsection 9.2.7 is a prime example of a computable metric space. Another prominent example is the infinite-dimensional separable Hilbert space ℓ_2 (cf. [9]). For more examples see [11, 44]. □

Propositions 9.4.12 and 9.4.13 tell us how to construct effectively admissible qcb-spaces from given ones.

Proposition 9.4.17 (Construction of effective qcb-spaces).
Let $\mathbf{X} = (X, \delta)$ *and* $\mathbf{Y} = (Y, \gamma)$ *be effective qcb-spaces. The following are effective qcb-spaces, where* X *and* Y *denote the* qcb_0*-spaces associated with* $\mathbf{X}$ *and* $\mathbf{Y}$*, respectively.*

1) $\mathbf{X} \times \mathbf{Y} := (\mathsf{X} \times \mathsf{Y}, \delta \boxtimes \gamma)$.
2) $\mathbf{X} \uplus \mathbf{Y} := (\{1\} \times \mathsf{X} \cup \{2\} \times \mathsf{Y}, \delta \boxplus \gamma)$.
3) $\mathbf{Y}^\mathbf{X} := (\mathsf{Y}^\mathsf{X}, [\delta \to \gamma])$.
4) $\mathbf{X} \wedge \mathbf{Y} := (\mathsf{X} \wedge \mathsf{Y}, \delta \wedge \gamma)$.
5) $\mathscr{O}(\mathbf{X}) := (\mathscr{O}(\mathsf{X}), \delta^\mathscr{O})$.
6) $\mathscr{A}_-(\mathbf{X}) := (\mathscr{A}_-(\mathsf{X}), \delta^{\mathscr{A}_-})$.
7) $\mathscr{A}_+(\mathbf{X}) := (\mathscr{A}_+(\mathsf{X}), \delta^{\mathscr{A}_+})$.

8) $\mathscr{K}(\mathbf{X}) := (\mathscr{K}(\mathsf{X}), \delta^{\mathscr{K}})$.
9) $\mathscr{K}_{\mathrm{Viet}}(\mathbf{X}) := (\mathscr{K}_{\mathrm{Viet}}(\mathsf{X}), \delta^{\mathscr{K}_{\mathrm{Viet}}})$.
10) $\mathbf{X}|^Z := (Z, \delta|^Z)$ *for* $Z \subseteq \mathbf{X}$*, where* $\delta|^Z$ *is the subspace representation.*

In Proposition 9.4.17 we have deliberately used qcb-spaces rather than sets as first component: by Propositions 9.4.12 and 9.4.13 they are the qcb-spaces associated with the represented spaces in (1) to (9). The qcb-space associated with $\mathbf{X}|^Z$ in (10) carries the subsequential topology on Z. The category EffQCB has nice closure properties.

Theorem 9.4.18. *The category* EffQCB *of effective qcb-spaces and total computable functions is cartesian closed, finitely complete and finitely cocomplete.*

9.4.7 Basic Computable Functions on Effective Qcb-Spaces

We present some basic computable functions on effective qcb-spaces. More can be found in A. Pauly's paper [32].

Proposition 9.4.19 ([32]).
Let $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ *be effective qcb-spaces. The following functions are computable:*

1) $pr_1 \colon \mathbf{X} \times \mathbf{Y} \to \mathbf{X}$, $(x,y) \mapsto x$, $pr_2 \colon \mathbf{X} \times \mathbf{Y} \to \mathbf{Y}$, $(x,y) \mapsto y$.
2) $eval \colon \mathbf{Y}^{\mathbf{X}} \times \mathbf{X} \to \mathbf{Y}$, $(f,x) \mapsto f(x)$.
3) $curry \colon \mathbf{Y}^{\mathbf{Z}\times\mathbf{X}} \to (\mathbf{Y}^{\mathbf{X}})^{\mathbf{Z}}$, $(curry(f)(z))(x) := f(z,x)$.
4) $uncurry \colon (\mathbf{Y}^{\mathbf{X}})^{\mathbf{Z}} \to \mathbf{Y}^{\mathbf{Z}\times\mathbf{X}}$, $uncurry(h)(z,x) := h(z)(x)$.
5) $join \colon \mathbf{X}^{\mathbf{Z}} \times \mathbf{Y}^{\mathbf{Z}} \to (\mathbf{X} \times \mathbf{Y})^{\mathbf{Z}}$, $join(f,g)(z) := (f(z), g(z))$.
6) $^{\complement} \colon \mathscr{O}(\mathbf{X}) \to \mathscr{A}_-(\mathbf{X})$, $U \mapsto \mathbf{X} \setminus U$.
7) $^{\complement} \colon \mathscr{A}_-(\mathbf{X}) \to \mathscr{O}(\mathbf{X})$, $A \mapsto \mathbf{X} \setminus A$.
8) $\cap \colon \mathscr{O}(\mathbf{X}) \times \mathscr{O}(\mathbf{X}) \to \mathscr{O}(\mathbf{X})$, $(U,V) \mapsto U \cap V$.
9) $\cup \colon \mathscr{O}(\mathbf{X}) \times \mathscr{O}(\mathbf{X}) \to \mathscr{O}(\mathbf{X})$, $(U,V) \mapsto U \cup V$.
10) $\bigcup \colon \mathscr{O}(\mathbf{X})^{\mathbb{N}} \to \mathscr{O}(\mathbf{X})$, $(U_n)_n \mapsto \bigcup_{n\in\mathbb{N}} U_n$.
11) $\cup \colon \mathscr{K}(\mathbf{X}) \times \mathscr{K}(\mathbf{X}) \to \mathscr{K}(\mathbf{X})$, $(K,L) \mapsto K \cup L$.
12) $\cup \colon \mathscr{A}_+(\mathbf{X}) \times \mathscr{A}_+(\mathbf{X}) \to \mathscr{A}_+(\mathbf{X})$, $(A,B) \mapsto A \cup B$.
13) $\cap \colon \mathscr{K}(\mathbf{X}) \times \mathscr{A}_-(\mathbf{X}) \to \mathscr{K}(\mathbf{X})$, $(K,A) \mapsto K \cap A$.
14) $\times \colon \mathscr{O}(\mathbf{X}) \times \mathscr{O}(\mathbf{Y}) \to \mathscr{O}(\mathbf{X} \times \mathbf{Y})$, $(U,V) \mapsto U \times V$.
15) $\times \colon \mathscr{A}_-(\mathbf{X}) \times \mathscr{A}_-(\mathbf{Y}) \to \mathscr{A}_-(\mathbf{X} \times \mathbf{Y})$, $(A,B) \mapsto A \times B$.
16) $\iota \colon \mathbf{X} \to \mathscr{K}(\mathbf{X})$, $x \mapsto {\uparrow}\{x\}$.
17) $\iota \colon \mathbf{X} \to \mathscr{A}_+(\mathbf{X})$, $x \mapsto \mathrm{Cls}\{x\}$.
18) $preimage \colon \mathbf{Y}^{\mathbf{X}} \times \mathscr{O}(\mathbf{Y}) \to \mathscr{O}(\mathbf{X})$, $(f,V) \mapsto f^{-1}[V]$.
19) $image \colon \mathbf{Y}^{\mathbf{X}} \times \mathscr{K}(\mathbf{X}) \to \mathscr{K}(\mathbf{Y})$, $(f,K) \mapsto {\uparrow}f[K]$.
20) $\forall \colon \mathscr{O}(\mathbf{X} \times \mathbf{Y}) \times \mathscr{K}(\mathbf{X}) \to \mathscr{O}(\mathbf{Y})$, $(W,K) \mapsto \{y \in \mathbf{Y} \mid \forall x \in K.(x,y) \in W\}$.
21) $\exists \colon \mathscr{O}(\mathbf{X} \times \mathbf{Y}) \times \mathscr{A}_+(\mathbf{X}) \to \mathscr{O}(\mathbf{Y})$, $(W,A) \mapsto \{y \in \mathbf{Y} \mid \exists x \in A.(x,y) \in W\}$.

We emphasise that $\forall(W,K)$ and $\exists(W,A)$ are indeed open subsets of the qcb_0-space associated with $\mathbf{Y}$. Related results can be found in [12, 16, 17, 44].

9.4.8 Effective Hausdorff Spaces

Topological spaces X in which every converging sequence has a unique limit are called *sequentially Hausdorff*. This condition is equivalent to sequential continuity of the inequality test $\neq\colon \mathsf{X}\times\mathsf{X}\to\mathbb{S}$ on X which is defined by $\neq(x,y)=\top :\Longleftrightarrow x\neq y$. Every Hausdorff space is sequentially Hausdorff; sequentially Hausdorff spaces are T_1. This gives rise to the notion of an *effective Hausdorff space*: this is an effective qcb-space such that its inequality test $\neq$ is computable. A similar notion has been investigated by A. Pauly in [32]: he calls a represented space $\mathbf{X}$ *computably* T_2 if the inequality test is computable; its representation, however, is not required to be effectively admissible. Any computable metric space forms an effective Hausdorff space.

Proposition 9.4.20 ([32]). *The following functions are computable for an effective Hausdorff space* $\mathbf{X}$*:*

1) $\cap\colon \mathscr{K}(\mathbf{X})\times\mathscr{K}(\mathbf{X})\to\mathscr{K}(\mathbf{X}),\quad (K,L)\mapsto K\cap L.$
2) *singleton*$\colon \mathbf{X}\to\mathscr{A}_-(\mathbf{X}),\ x\mapsto\{x\}.$
3) *convert*$\colon \mathscr{K}(\mathbf{X})\to\mathscr{A}_-(\mathbf{X}),\ K\mapsto K.$
4) Graph$\colon \mathbf{X}^{\mathbf{Z}}\to\mathscr{A}_-(\mathbf{Z}\times\mathbf{X}),\ f\mapsto\{(z,f(z))\,|\,z\in\mathbf{Z}\}$ *is computable for any effective qcb-space* $\mathbf{Z}$*.*

We emphasise that the above functions map indeed into the indicated sets, whenever $\mathbf{X}$ is an effective Hausdorff space. Conversely, if $\mathbf{X}$ is an effective qcb-space such that one of the functions listed in Proposition 9.4.20 is well defined and computable, then the inequality test on $\mathbf{X}$ is computable, hence $\mathbf{X}$ is an effective Hausdorff space. This was shown by A. Pauly in [32].

The reader should be warned that effective Hausdorff spaces exist which are not Hausdorff in the topological sense, for example the one-point compactification of the Baire space. Nevertheless, the arguably more precise notion "effective sequential Hausdorff spaces" seems too clumsy. For second-countable spaces, however, sequential Hausdorffness coincides with Hausdorffness. K. Weihrauch has investigated and compared several notions of computable Hausdorff separation for second-countable computable topological spaces in [46].

9.4.9 Quasi-normal Qcb-Spaces

We present a subcategory of admissibly represented Hausdorff spaces that is particularly appropriate for investigating computability in Functional Analysis.

Normality[2] has the disadvantage of not being preserved by exponentiation in QCB. For example, the QCB-exponential $\mathbb{R}^{(\mathbb{R}^{\mathbb{R}})}$ is not normal, although $\mathbb{R}^{\mathbb{R}}$ and $\mathbb{R}$

[2] A T_1-space is called *normal* if any two disjoint closed sets A,B can be separated by two disjoint open sets U,V in the sense that $A\subseteq U$ and $B\subseteq V$ holds.

are even metrisable. So we introduce a substitute for normality in the realm of qcb-spaces called quasi-normality. A *quasi-normal* space is defined to be the sequential coreflection (see Subsection 9.3.3) of some normal Hausdorff space. We write QN for the full subcategory of QCB consisting of all quasi-normal qcb-spaces. In contrast to normal qcb-spaces, QN has excellent closure properties.

Theorem 9.4.21 (Closure properties of quasi-normal spaces).
The category QN *of quasi-normal qcb-spaces is an exponential ideal of* QCB *and thus cartesian closed. Moreover,* QN *has all countable limits and all countable colimits. Countable limits and function spaces are inherited from* QCB.

Hence the QCB-exponential $\mathbb{R}^{(\mathbb{R}^{\mathbb{R}})}$ lives inside QN. Relevant examples for non-metrisable QN-spaces are the space of test functions $\mathscr{D}$ and its dual $\mathscr{D}'$, which is known as the space of distributions (= generalised functions). Computability on distributions has been investigated by K. Weihrauch and N. Zhong in [48].

The famous Tietze-Urysohn Extension Theorem states that any real-valued function defined on a closed subspace of a normal topological space can be extended to a continuous function defined on the whole space. Quasi-normal qcb-spaces admit continuous extendability of continuous functions defined on subspaces that are functionally closed. *Functionally closed* subsets are exactly the preimages of 0 under continuous real-valued functions. We state a uniform version.

Theorem 9.4.22 (Uniform Extension Theorem for QN-spaces).
Let X *be a functionally closed subspace of a quasi-normal qcb-space* Y. *Then there is a continuous functional* $E\colon \mathbb{R}^{\mathsf{X}} \to \mathbb{R}^{\mathsf{Y}}$ *satisfying* $E(f)(x) = f(x)$ *for all continuous functions* $f\colon \mathsf{X} \to \mathbb{R}$ *and all* $x \in \mathsf{X}$.

A qcb-space X is quasi-normal if, and only if, X embeds sequentially into $\mathbb{R}^{(\mathbb{R}^{\mathsf{X}})}$ via the embedding $\eta_{\mathsf{X},\mathbb{R}}\colon x \mapsto (h \mapsto h(x))$. Hence $\mathbb{R}$ plays a similar role for quasi-normal spaces as the Sierpiński space does for sequential spaces. Generalising the ideas of Subsection 9.4.6, we call a represented space $\mathbf{X} = (X, \delta)$ *effectively quasi-normal* if τ_δ is a T_0-topology and δ is computably equivalent to the representation $\delta^{\mathbb{R}}$ defined by

$$\delta^{\mathbb{R}}(p) = x \;:\Longleftrightarrow\; \eta_{(X,\tau_\delta),\mathbb{R}}(x) = [[\delta \to \rho_{\mathbb{R}}] \to \rho_{\mathbb{R}}](p)\,.$$

The operator $\delta \mapsto \delta^{\mathbb{R}}$ behaves like the operator $\delta \mapsto \delta^{\mathbb{S}}$ from Subsection 9.4.4.

Proposition 9.4.23. *Let* δ *and* γ *be quotient representations of* QN*-spaces* X *and* Y, *respectively.*

1) *Any* (δ,γ)*-computable total function is* $(\delta^{\mathbb{R}},\gamma^{\mathbb{R}})$*-computable.*
2) *Any* (δ,γ)*-continuous total function is* $(\delta^{\mathbb{R}},\gamma^{\mathbb{R}})$*-continuous.*
3) *The spaces* $(\mathsf{X},\delta^{\mathbb{R}})$ *and* $(\mathsf{Y},\gamma^{\mathbb{R}})$ *are effectively quasi-normal.*
4) *Any effective quasi-normal space is an effective qcb-space.*
5) *Any computable metric space equipped with its Cauchy representation gives rise to an effective quasi-normal space.*

More information can be found in [38].

An interesting subclass of quasi-normal spaces is the class of *co-Polish* Hausdorff spaces. These are those Hausdorff qcb-spaces which have an admissible representation with a locally compact domain. M. de Brecht has proven that co-Polish Hausdorff spaces are exactly those Hausdorff qcb-spaces X for which $\mathscr{O}(\mathsf{X})$, the lattice of open subsets equipped with the Scott topology (cf. Subsection 9.4.3.1), is quasi-Polish [13]. This fact motivates the name "co-Polish". D. Kunkle showed that any Silva space is co-Polish (cf. [27]). Silva spaces (cf. [39]) form an important subclass of locally convex vector spaces.

9.5 Relationship to Other Relevant Categories

There are several cartesian closed categories relevant to Computable Analysis into which QCB_0 embeds as a subcategory that inherits binary products and function spaces. As examples we discuss represented spaces, equilogical spaces, limit spaces, filter spaces, sequential spaces, compactly generated spaces and core compactly generated spaces.

9.5.1 Represented Spaces

Assuming a mild axiom of choice, QCB_0 embeds into Rep as a subccc (meaning that QCB_0 inherits the cartesian closed structure). The inclusion functor maps a qcb_0-space X to $(\mathsf{X}, \delta_{\mathsf{X}})$, where δ_{X} is a chosen admissible representation of X. This functor has a left adjoint (cf. [30]): it maps a represented space $(Y, \gamma) \in \mathsf{Rep}$ to the Kolmogorov quotient of the qcb-space (Y, τ_γ), where τ_γ denotes the final topology. Similarly, EffQCB is a full reflective subcategory of $\mathsf{Rep}_{\mathrm{eff}}$; the reflection functor maps (Y, γ) to the effective qcb-space $(\mathsf{KQ}(Y, \tau_\gamma), \gamma^{\mathbb{S}})$.

9.5.2 Equilogical Spaces

The category Top of topological spaces is well known not to be cartesian closed. For example, there exists no exponential to the metrisable space $\mathbb{R}^{\mathbb{R}}$ and $\mathbb{R}$. Note that the QCB-exponential $\mathbb{R}^{(\mathbb{R}^{\mathbb{R}})}$ does not form an exponential in the category Top, because the evaluation function is not topologically continuous (albeit sequentially continuous). However, there exist cartesian closed supercategories of Top. We mention M. Hyland's category of filter spaces (cf. [23, 25]) and D. Scott's category Equ of equilogical spaces (cf. [5]) as important examples.

An *equilogical space* is a topological T_0-space together with an equivalence relation on that space. It is called *countably based* if the topological space has a count-

able base. An *equivariant* map between two equilogical spaces is a function between the induced equivalence classes of the equilogical spaces that is realized by a continuous function between the underlying topological spaces. The category Equ of equilogical spaces and equivariant maps as well as its full subcategory ωEqu of countably based equilogical spaces are locally cartesian closed [3, 5].

Equilogical spaces form a generalisation of represented spaces. Any represented space (X, δ) can be seen as a countably based equilogical space: the underlying topological space is just the domain of the representation and the equivalence relation is given by $p \equiv q :\Longleftrightarrow \delta(p) = \delta(q)$. The ensuing inclusion functor $\mathsf{Rep} \hookrightarrow \omega\mathsf{Equ}$ has the disadvantage of not preserving function spaces (cf. [3, 4]). However, composition with the functor $\mathsf{QCB}_0 \hookrightarrow \mathsf{Rep}$ from Subsection 9.5.1 yields an inclusion functor $\mathsf{QCB}_0 \hookrightarrow \omega\mathsf{Equ}$ that does preserve binary products and function spaces. So QCB_0 lives inside ωEqu as a subccc. M. Menni and A. Simpson have exhibited QCB_0 as the largest common subcategory of ωEqu and Top that is cartesian closed and contains all T_0-spaces with a countable base (cf. [30]).

9.5.3 Limit Spaces

A reasonable notion of admissible representation exists for other classes of spaces besides the classes of the topological spaces discussed before. We concentrate on limit spaces as the most useful example. However, the largest class of spaces which can be equipped with an admissible multirepresentation is the class of weak limit spaces with a countable limit base (cf. [37]). The corresponding category ωWLim is equivalent to the category ωPFil of proper filter spaces with a countable basis studied by M. Hyland in [25].

A *limit space* ([28]) equips a set X with a *convergence relation* $\to$ between sequences $(x_n)_n$ and points x of X. If $(x_n)_n \to x$, then one says that $(x_n)_n$ converges to x in the space $(X, \to)$ and that x is a limit of the sequence $(x_n)_n$. The convergence relation of a limit space is required to satisfy the following properties:

1) any constant sequence $(x)_n$ converges to x;
2) any subsequence of a sequence converging to x converges to x;
3) a sequence converges to x whenever any of its subsequences has a subsequence converging to x.

Clearly, the convergence relation of a topological space satisfies these axioms.

A partial function between limit spaces is called *continuous* if it preserves convergent sequences (cf. Subsection 9.3.2). The category Lim of limit spaces and total continuous functions is locally cartesian closed, in contrast to the category Seq of sequential spaces, which is merely cartesian closed. Like Seq, it is countably complete and countably cocomplete. The category Seq is a full reflective subcategory of Lim (cf. [25]). The inclusion functor maps a sequential space Y to the limit space that has the same convergence relation. The reflection functor sends a limit space X to the sequential space that carries the topology of *sequentially open* subsets of X;

these are exactly the complements of subsets closed under forming limits of converging sequences (cf. Subsection 9.3.2). If the resulting sequential space induces the same convergence relation as X, then X is called *topological*.

Example 9.5.1 (A limit space that is not topological).
We equip the set $Z := \{(1,0)\} \cup (\{2\} \times \mathbb{N}) \cup (\{3\} \times \mathbb{N})$ with the following convergence relation $\to$:

$$\begin{aligned}
(z_n)_n \to (1,0) &:\Longleftrightarrow \forall m.\exists \tilde{n}.\forall n \geq \tilde{n}.\, z_n \in \{(1,0),(2,j) \mid j \geq m\} \\
(z_n)_n \to (2,a) &:\Longleftrightarrow \forall m.\exists \tilde{n}.\forall n \geq \tilde{n}.\, z_n \in \{(2,a),(3,j) \mid j \geq m\} \\
(z_n)_n \to (3,a) &:\Longleftrightarrow \exists \tilde{n}.\forall n \geq \tilde{n}.\, z_n = (3,a).
\end{aligned}$$

The resulting space Z is indeed a limit space. Any sequentially open subset V of Z with $(1,0) \in V$ contains the sequence $(3,n)_n$ eventually. Therefore the reflection functor maps Z to a sequential space in which $(3,n)_n$ converges to $(1,0)$. Hence Z is not topological. □

A multirepresentation δ for a limit space X is called *continuous* if for any sequence $(p_n,x_n)_n$ in $\mathrm{Graph}(\delta)$ such that $(p_n)_n$ converges to p_0 in $\mathbb{N}^{\mathbb{N}}$ the sequence $(x_n)_n$ converges to x_0 in X. It is called *admissible* for X if it is continuous and every continuous multirepresentation ϕ of X is continuously translatable into δ, meaning that the identity function is (ϕ,δ)-continuous. If two elements $x \neq x'$ have a common name under a continuous multirepresentation, then both elements converge to each other as constant sequences. An admissible multirepresentation of X is single-valued (i.e., an ordinary representation) if, and only if, no such elements $x \neq x'$ exist in X. This notion of admissibility extends topological admissibility from Definition 9.3.4 in the sense that a representation is topologically admissible for a qcb_0-space Y if, and only if, it is admissible for Y viewed as a limit space.

Example 9.5.2 (Lim*-admissible representation).*
An admissible representation ζ for the limit space Z in Example 9.5.1 is given by $\zeta(0^\omega) := (1,0)$, $\zeta(00^a1^\omega) := (2,a) =: \zeta((a+1)0^\omega)$, $\zeta((a+1)0^b1^\omega) := (3,b)$ for all $a,b \in \mathbb{N}$. □

Like Lim, the full subcategory $\omega\mathsf{Lim}$ of limit spaces with an admissible multirepresentation is locally cartesian closed, countably complete and countably cocomplete. The category QCB is a full reflective subcategory of $\omega\mathsf{Lim}$; the corresponding reflection functor is just the restriction of the aforementioned functor from Lim to Seq. On the other hand, $\omega\mathsf{Lim}$ is a full reflective subcategory of M. Hyland's category $\omega\mathsf{PFil}$ of countably based proper filter spaces.

We now discuss the Λ-space, which plays the role of the Sierpiński space in the realm of limit spaces.

Example 9.5.3 (The Λ*-space).*
The *Λ-space* $\mathbb{L}$ has as its underlying set $\{\bot,\top,\uparrow\}$. Its convergence relation is defined by: $(b_n)_n \to_{\mathbb{L}} b_\infty$ iff $b_\infty \neq \top$ or $b_n \neq \bot$ for almost all n. Hence the restriction

of $\mathbb{L}$ to $\{\bot,\top\}$ is the limit space version of the Sierpiński space $\mathbb{S}$. An admissible multirepresentation $\rho_{\mathbb{L}}$ for $\mathbb{L}$ is defined by

$$\rho_{\mathbb{L}}[p] := \begin{cases} \{\bot,\uparrow\} \text{ if } p = 0^\omega \\ \{\top,\uparrow\} \text{ otherwise} \end{cases}$$

for all $p \in \mathbb{N}^{\mathbb{N}}$. The Λ-space has the property that any limit space embeds into $\mathbb{L}^{\mathbb{L}^{\mathsf{X}}}$ via the map $\eta_{\mathsf{X},\mathbb{L}} : x \mapsto (h \mapsto h(x))$. Remember that any sequential space Y embeds sequentially into $\mathbb{S}^{\mathbb{S}^{\mathsf{Y}}}$ via an analogous map (cf. Proposition 9.4.4). □

The Λ-space can be employed to introduce the notion of an *effectively* Lim*-admissible (multi-)representation* for a limit space. The definition uses ideas similar to the ones in Subsection 9.4.5. If δ lifts convergent sequences of X, then $\delta^{\mathbb{L}}$ defined by

$$\delta^{\mathbb{L}}[p] \ni x \ :\Longleftrightarrow\ \eta_{\mathsf{X},\mathbb{L}}(x) \in [[\delta \to \rho_{\mathbb{L}}] \to \rho_{\mathbb{L}}][p]$$

is an admissible multirepresentation of X. A multirepresented space $\mathbf{X} = (X,\delta)$ is called an *effective limit space* if δ is computably equivalent to $\delta^{\mathbb{L}}$ (cf. [33, 34]).

Example 9.5.4 (Effective limit space).
For effective limit spaces $\mathbf{X},\mathbf{Y}$, the space $\mathbf{P}(\mathbf{X},\mathbf{Y})$ of partial continuous functions from $\mathbf{X}$ to $\mathbf{Y}$ from Example 9.2.16 is an effective limit space. But in general it is not an effective qcb-space (even if one allowed multirepresentations in the notion of an effective qcb-space). □

A further advantage of effective limit spaces over effective qcb-spaces is that the operator $\delta \mapsto \delta^{\mathbb{L}}$ preserves computability even of partial functions.

Proposition 9.5.5. *Let δ and γ be multirepresentations of limit spaces X,Y lifting converging sequences.*

1) *Any (δ,γ)-computable partial function is $(\delta^{\mathbb{L}},\gamma^{\mathbb{L}})$-computable.*
2) *Any (δ,γ)-continuous partial function is $(\delta^{\mathbb{L}},\gamma^{\mathbb{L}})$-continuous.*
3) *The multirepresentations $\delta^{\mathbb{L}}$ and $\gamma^{\mathbb{L}}$ are effectively* Lim*-admissible.*
4) *Any effective qcb-space constitutes an effective limit space.*

The category EffLim of effective limit spaces and total computable functions is locally cartesian closed and has finite limits and finite colimits. It contains EffQCB as subccc.

More information can be found in [25, 29, 30, 33, 34].

9.5.4 Cartesian Closed Subcategories of Topological Spaces

Although the category Top of topological spaces is not cartesian closed, it contains various subcategories which are cartesian closed. Several of them were investigated by M. Escardó, J. Lawson and A. Simpson in [18]. Besides the category Seq of

sequential spaces, we mention the categories kTop of compactly generated spaces and CCG of core compactly generated spaces.

A topological space Z is called *compactly generated* (or a *Kelley space*) if any subset U is open in Z, whenever $p^{-1}[U]$ is open in K for every compact Hausdorff space K and every continuous function $p\colon \mathsf{K} \to \mathsf{Z}$. For example, any directed-complete poset endowed with the Scott topology is compactly generated. Compactly generated Hausdorff spaces are known as *k-spaces* and play an important role in algebraic topology.

Even larger is the category CCG of core compactly generated spaces. *Core compactly generated spaces* arise as all topological quotients of core compact (= exponentiable) topological spaces. None of the inclusion functors of these categories into Top preserves finite products. The category QCB of qcb-spaces lives inside Seq, kTop and CCG as a subccc. In fact, QCB, Seq, kTop and CCG form an increasing chain of cartesian closed categories such that any smaller category inherits binary products and function spaces from any larger one. Details can be found in [18].

Acknowledgements I thank Vasco Brattka, Matthew de Brecht, Peter Hertling and Thomas Streicher for valuable discussions.

References

1. Battenfeld, I., Schröder, M., Simpson, A.: Compactly generated domain theory. Mathematical Structures in Computer Science **16**(2), 141–161 (2006)
2. Battenfeld, I., Schröder, M., Simpson, A.: A convenient category of domains. Electronic Notes in Theoretical Computer Science **172**, 69–99 (2007)
3. Bauer, A.: The realizability approach to computable analysis and topology. School of Computer Science, Carnegie Mellon University, Pittsburgh (2000)
4. Bauer, A.: A relationship between equilogical spaces and type two effectivity. In: S. Brooks, M. Mislove (eds.) Seventeenth Conference on the Mathematical Foundations of Programming Semantics, *Electronic Notes in Theoretical Computer Science*, vol. 45. Elsevier, Amsterdam (2001), MFPS 2001, Aarhus, Denmark May 23–26, 2001
5. Bauer, A., Birkedal, L., Scott, D.S.: Equilogical spaces. Theoretical Computer Science **315**, 35–59 (2004)
6. Brattka, V.: Recursive characterization of computable real-valued functions and relations. Theoretical Computer Science **162**, 45–77 (1996)
7. Brattka, V.: Recursive and computable operations over topological structures. Ph.D. thesis. Department of Computer Science, University of Hagen, Hagen, Germany (1998)
8. Brattka, V.: Computability over topological structures. In: S.B. Cooper, S.S. Goncharov (eds.) Computability and Models, pp. 93–136. Kluwer Academic Publishers, New York (2003)
9. Brattka, V., Dillhage, R.: On computable compact operators on computable Banach spaces with bases. Mathematical Logic Quarterly **53**(4–5), 345–364 (2007)
10. Brattka, V., Hertling, P.: Feasible real random access machines. Journal of Complexity **14**(4), 490–526 (1998)
11. Brattka, V., Hertling, P., Weihrauch, K.: A tutorial on computable analysis. In: S.B. Cooper, B. Löwe, A. Sorbi (eds.) New Computational Paradigms: Changing Conceptions of What Is Computable, pp. 425–491. Springer, New York (2008)
12. Brattka, V., Presser, G.: Computability on subsets of metric spaces. Theoretical Computer Science **305**, 43–76 (2003)

13. de Brecht, M.: Quasi-Polish spaces. Ann. Pure Appl. Logic **164**(3), 356–381 (2013)
14. de Brecht, M., Schröder, M., Selivanov, V.: Base-complexity classification of QCB_0–spaces. Computability **5**(1), 75–102 (2016)
15. Engelking, R.: General Topology, *Sigma Series in Pure Mathematics*, vol. 6. Heldermann, Berlin (1989)
16. Escardó, M.: Synthetic topology of data types and classical spaces. Electronic Notes in Theoretical Computer Science **87**, 21–165 (2004)
17. Escardó, M.: Exhaustible sets in higher-type computation. Logical Methods in Computer Science **4**(3), 3:3, 37 (2008)
18. Escardó, M., Lawson, J., Simpson, A.: Comparing Cartesian closed categories of (core) compactly generated spaces. Topology and its Applications **143**(1-3), 105–145 (2004)
19. Escardó, M.H.: PCF extended with real numbers. Theoretical Computer Science **162**, 79–115 (1996)
20. Grzegorczyk, A.: On the definitions of computable real continuous functions. Fundamenta Mathematicae **44**, 61–71 (1957)
21. Hauck, J.: Berechenbare reelle Funktionen. Zeitschrift für Mathematische Logik und Grundlagen der Mathematik **19**, 121–140 (1973)
22. Hauck, J.: Berechenbarkeit in topologischen Räumen mit rekursiver Basis. Zeitschrift für Mathematische Logik und Grundlagen der Mathematik **27**, 473–480 (1981)
23. Heckmann, R.: On the relationship between filter spaces and equilogical spaces (1998), manuscript
24. Hertling, P.: A comparison of certain representations of regularly closed sets. In: V. Brattka, M. Schröder, K. Weihrauch (eds.) CCA 2002 Computability and Complexity in Analysis, *Electronic Notes in Theoretical Computer Science*, vol. 66. Elsevier, Amsterdam (2002), 5th International Workshop, CCA 2002, Málaga, Spain, July 12–13, 2002
25. Hyland, J.: Filter spaces and continuous functionals. Annals of Mathematical Logic **16**, 101–143 (1979)
26. Kreitz, C., Weihrauch, K.: Theory of representations. Theoretical Computer Science **38**, 35–53 (1985)
27. Kunkle, D., Schröder, M.: Some examples of non-metrizable spaces allowing a simple type-2 complexity theory. In: V. Brattka, L. Staiger, K. Weihrauch (eds.) Proceedings of the 6th Workshop on Computability and Complexity in Analysis, *Electronic Notes in Theoretical Computer Science*, vol. 120, pp. 111–123. Elsevier, Amsterdam (2005), 6th International Workshop, CCA 2004, Wittenberg, Germany, August 16–20, 2004
28. Kuratowski, K.: Topology, Volume 1. Academic Press, New York (1966)
29. Longley, J., Normann, D.: Higher-Order Computability, 1st edn. Springer (2015)
30. Menni, M., Simpson, A.: Topological and limit-space subcategories of countably-based equilogical spaces. Mathematical Structures in Computer Science **12**(6), 739–770 (2002)
31. Müller, N.T.: The iRRAM: Exact arithmetic in C++. In: J. Blanck, V. Brattka, P. Hertling (eds.) Computability and Complexity in Analysis, *Lecture Notes in Computer Science*, vol. 2064, pp. 222–252. Springer, Berlin (2001), 4th International Workshop, CCA 2000, Swansea, UK, September 2000
32. Pauly, A.: On the topological aspects of the theory of represented spaces. Computability **5**(2), 159–180 (2016)
33. Schröder, M.: Admissible representations for continuous computations. Ph.D. thesis. Fachbereich Informatik, FernUniversität Hagen (2002)
34. Schröder, M.: Effectivity in spaces with admissible multirepresentations. Mathematical Logic Quarterly **48**(Suppl. 1), 78–90 (2002)
35. Schröder, M.: Extended admissibility. Theoretical Computer Science **284**(2), 519–538 (2002)
36. Schröder, M.: Spaces allowing type-2 complexity theory revisited. Mathematical Logic Quarterly **50**(4,5), 443–459 (2004)
37. Schröder, M.: On the relationship between filter spaces and weak limit spaces. Journal of Universal Computer Science **14**(6), 996–1015 (2008)
38. Schröder, M.: An effective Tietze-Urysohn theorem for QCB-spaces. Journal of Universal Computer Science **15**(6), 1317–1336 (2009)

39. Sebastião e Silva, J.: Su certe classi di spazi localmente convessi importanti per le applicazioni. Rend. Mat. e Appl. (5) **14**, 388–410 (1955)
40. Simpson, A.K.: Towards a convenient category of topological domains. In: Proceedings of the 13th ALGI Workshop, RIMS, Kyoto University (2003)
41. Taylor, P.: A lambda calculus for real analysis. Journal of Logic and Analysis **2**, Paper 5, 115 (2010)
42. Turing, A.M.: On computable numbers, with an application to the "Entscheidungsproblem". A correction. Proceedings of the London Mathematical Society **43**(2), 544–546 (1938)
43. Weihrauch, K.: Computability, *EATCS Monographs on Theoretical Computer Science*, vol. 9. Springer, Berlin (1987)
44. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)
45. Weihrauch, K.: Computational complexity on computable metric spaces. Mathematical Logic Quarterly **49**(1), 3–21 (2003)
46. Weihrauch, K.: Computable separation in topology, from T_0 to T_2. Journal of Universal Computer Science **16**(18), 2733–2753 (2010)
47. Weihrauch, K., Grubba, T.: Elementary computable topology. Journal of Universal Computer Science **15**(6), 1381–1422 (2009)
48. Zhong, N., Weihrauch, K.: Computability theory of generalized functions. Journal of the Association for Computing Machinery **50**(4), 469–505 (2003)
49. Ziegler, M.: Computable operators on regular sets. Mathematical Logic Quarterly **50**(4,5), 392–404 (2004)

Chapter 10
Bishop-Style Constructive Reverse Mathematics

Hannes Diener and Hajime Ishihara

Abstract We give a self-contained overview of the current state of Constructive Reverse Mathematics.

10.1 Introduction

One of the reasons Bishop was able to develop large parts of analysis constructively [13, 14] without getting drawn into philosophical arguments as Brouwer did before him ("Grundlagenstreit") was that he chose an agnostic approach to the principle of excluded middle (PEM),[1] and other principles Brouwer accepted, such as the principle of continuous choice and bar induction. PEM, of course, states that for every formula φ we have $\varphi \vee \neg\varphi$. Over intuitionistic logic this is equivalent to double negation elimination (DNE), which states that for every formula φ we have $\neg\neg\varphi \to \varphi$.

Working only over intuitionistic logic and not assuming any of Brouwer's additional principles meant that Bishop's mathematics stayed compatible with both classical mathematics (CLASS) as well as Brouwer's intuitionism (INT) [37, 29]. More than that, as it later became clear [22], BISH is also compatible with Russian Recursive Mathematics (RUSS) [51].

However, Bishop's minimal approach meant that in his mathematics many common principles and theorems are neither provable nor is their negation provable. Of

Hannes Diener
School of Mathematics and Statistics, University of Canterbury, Christchurch, New Zealand
e-mail: hannes.diener@canterbury.ac.nz

Hajime Ishihara
School of Information Science, Japan Advanced Institute of Science and Technology, Nomi, Japan
e-mail: ishihara@jaist.ac.jp

[1] PEM is also often referred to as the "law of excluded middle" (LEM) or its Latin name "tertium non datur".

V. Brattka, P. Hertling (eds.), *Handbook of Computability and Complexity in Analysis*,
Theory and Applications of Computability, https://doi.org/10.1007/978-3-030-59234-9_10

course, since Gödel we know that such statements are basically always unavoidable, however in BISH this happens not just for (arguably) artificial statements, but for very basic ones. It is the goal of constructive reverse mathematics to find some order in the multitude of principles and results that occupy this middle ground in BISH.

There are two similar *reverse* programs, which, while coming from very different directions, result in a—at least in parts—similar hierarchy. Maybe the best-known and most developed of all reverse approaches is "Simpson-style" reverse mathematics [64] (initiated by Friedman [31]), whose goal is to examine which set existence axioms need to be added to classical second-order arithmetic to prove theorems in mathematics (where objects are coded by natural numbers). A much younger[2] reverse approach is the theory of Weihrauch reducibility [15, 16]. The question there is: "which theorems can be computably transformed into others?"

One aspect that contrasts (Bishop-style) constructive reverse mathematics with these two other approaches is the fact that, as common in Bishop-style mathematics, we assume the following axioms of choice throughout the paper.[3]

The axiom of countable choice (AC_0):

$$\forall n \exists y \in Y A(n,y) \to \exists f \in Y^{\mathbf{N}} \forall n A(n, f(n)) \ .$$

The axiom of dependent choice (DC):

$$\forall x \in X \exists y \in X A(x,y) \to \forall x \in X \exists f \in X^{\mathbf{N}} [f(0) = x \wedge \forall n A(f(n), f(n+1))] \ .$$

The axiom of unique choice (AC!):

$$\forall x \in X \exists! y \in Y A(x,y) \to \exists f \in Y^X \forall x \in X A(x, f(x)) \ .$$

10.2 Preliminaries

Throughout this chapter, we follow the notational conventions in [67, 66]: m, n, i, j, k are supposed to range over $\mathbf{N}$, a, b, c over the set $\mathbf{N}^*$ of finite sequences of $\mathbf{N}$, and $\alpha, \beta, \gamma, \delta$ over $\mathbf{N}^{\mathbf{N}}$ (Baire space). The empty sequence is denoted by $\langle\rangle$.

For $\alpha, \beta \in \mathbf{N}^{\mathbf{N}}$, let $\alpha \# \beta \Leftrightarrow \exists n(\alpha n \neq \beta n)$, and let $\mathbf{0} = \lambda x.0$.

---

[2] The initial idea and definition of Weihrauch reducibility can be traced back to unpublished papers by Weihrauch in the 1990s.

[3] There are some constructivists who have argued in favour of avoiding the axioms of countable choice and dependent choice as much as possible, such as Richman [60]. As for constructive *reverse* mathematics, there has been, mostly recently, more and more work [11, 43, 45] which treats choice principles explicitly. That is, instead of showing that P_1 and P_2 are equivalent using countable choice, they are of the form P_1 is equivalent to $P_2 + \mathrm{AC}'$, where AC' is some appropriate weakening of choice.

If $a, b \in \mathbf{N}^*$ we denote the *concatenation* of the two sequences by $a * b$. Similarly we define $a * \alpha$ for $a \in \mathbf{N}^*$ and $\alpha \in \mathbf{N}^{\mathbf{N}}$.

Let $|a|$ denote the length of a finite sequence a; $\overline{a}(n)$ and $\overline{\alpha}(n)$ denote the initial segments of a and α of length n, respectively, where $n \leqslant |a|$. We write $a \preceq b$ if a is a prefix of b, that is if there is $m \leqslant |b|$ such that $\overline{b}(m) = a$. A *tree* is a subset T of $\{0,1\}^*$ such that $\langle\rangle \in T$ and if $b \in T$ and $a \preceq b$, then $a \in T$ for each a and b. A tree is *infinite* if for each n there exists $a \in T$ with $|a| = n$. We say that a sequence γ *is a branch of a tree* T or that T *has* γ *as a branch* if $\overline{\gamma}(n) \in T$ for each n.

A formula φ is called *decidable* if $\varphi \vee \neg\varphi$. In the absence of PEM not every formula is decidable, and in particular set membership might not be. If $A \subset X$ is such that $\forall x \in X\,[x \in A \vee x \notin A]$, however, we call A *detachable from* X. If the ambient set X is clear from the context, we will sometimes call A decidable. Because of this problem of membership possibly being not decidable, we also have to distinguish between a set S being non-empty—that is $\neg(S = \emptyset)$, and a set being *inhabited*—that is $\exists x(x \in S)$. Being a positive notion, the latter is most commonly used by constructivists.

Just as in traditional recursion theory, from which we also borrow the naming convention, we are also interested in predicates which, while not fully decidable, still have some sort of computational description. A predicate A on X is called Σ_1 if there exists a decidable predicate S on $X \times \mathbf{N}$ such that

$$\forall x \in X\,[A(x) \leftrightarrow \exists n(S(x,n))]\ ;$$

A it is called Π_1 if

$$\forall x \in X\,[A(x) \leftrightarrow \forall n(S(x,n))]\ .$$

A set A is *countable* if there is a surjection $f : \mathbf{N} \to A$.

A *real number* is a sequence $(p_n)_{n\geqslant 0}$ of rationals such that

$$\forall mn\left(|p_m - p_n| < 2^{-m} + 2^{-n}\right)\ .$$

We shall write $\mathbf{R}$ for the set of real numbers as usual. The *ordering relation* $<$ between real numbers $x = (p_n)_{n\geqslant 0}$ and $y = (q_n)_{n\geqslant 0}$ is defined by

$$x < y \iff \exists n\left(2^{-n+2} < q_n - p_n\right)\ .$$

This ordering relation satisfies the usual properties such as

$$\neg(x < y \wedge y < x),\ \text{and}\ x < y \to x < z \vee z < y\ ,$$

for $x, y, z \in \mathbf{R}$.

We define the *apartness* #, the *equality* $=$, and the ordering relation $\leqslant$ between real numbers x and y by $x \# y \Leftrightarrow (x < y \vee y < x)$, $x = y \Leftrightarrow \neg(x \# y)$, and $x \leqslant y \Leftrightarrow \neg(y < x)$.

It is straightforward to show that # is an apartness relation, that is, it satisfies

$$x \# y \leftrightarrow y \# x \qquad \text{and} \qquad x \# y \to x \# z \vee z \# y\ ,$$

and that $=$ is reflexive, symmetric, and transitive; that is, an equivalence relation.

A very basic result that allows one to jump between binary sequences and real numbers, and which leads to matching characterisations of many of the principles in Sections 10.3 and 10.4 is the following.

Lemma 10.2.1. *For each α, there exists $x \in \mathbf{R}$ such that*

$$\alpha \# \mathbf{0} \leftrightarrow x \# 0 \, .$$

Conversely, for each $x \in \mathbf{R}$, there exists α such that

$$x \# 0 \leftrightarrow \alpha \# \mathbf{0} \ .$$

This lemma might seem strange to a classically minded reader, who we would like to remind that constructively apartness # is neither decidable for elements of $\mathbf{R}$ nor is it decidable for elements of $\mathbf{N}^{\mathbf{N}}$.

Let S be an inhabited subset of $\mathbf{R}$. Then $z \in \mathbf{R}$ is an *upper bound* of S if $s \leqslant z$ for each $s \in S$; *least upper bound* of S if z is an upper bound and $z \leqslant y$ for each upper bound y of S, and we then write $z = \mathrm{lub} S$; *supremum* of S if it is an upper bound of S and for each $\varepsilon > 0$ there exists a real number $s \in S$ with $s > z - \varepsilon$, and we then write $z = \sup S$. One can show that an inhabited subset S of $\mathbf{R}$ that is bounded above has a supremum if and only if for all a and b with $a < b$, either there exists $s \in S$ such that $a < s$ or $s < b$ for all $s \in S$. Notice that, even though the definition of a supremum and that of a least upper bound are classically equivalent for the real line, being a supremum is, constructively, a stronger requirement than being a least upper bound, as one can see from Proposition 10.3.1. Lower bounds, greatest lower bounds (glb), and infima (inf) are defined analogously.

A *metric space* is a set X equipped with a *metric* $d : X \times X \to \mathbf{R}$ such that

(M1) $d(x,y) = 0 \leftrightarrow x = y$,
(M2) $d(x,y) = d(y,x)$,
(M3) $d(x,y) \leqslant d(x,z) + d(z,y)$,

for all $x, y, z \in X$. For $x, y \in X$, we write $x \# y$ for $0 < d(x,y)$.

A sequence $(x_n)_{n \geqslant 0}$ of X *converges to* $x \in X$, or in symbols $x_n \to x$ if

$$\forall k \exists N_k \forall n \geqslant N_k [d(x_n, x) < 2^{-k}] \, ,$$

and is a *Cauchy sequence* if

$$\forall k \exists N_k \forall mn \geqslant N_k [d(x_m, x_n) < 2^{-k}] \, .$$

A metric space is *complete* if every Cauchy sequence converges.

A metric space X is *totally bounded* if for all k there exist $x_0, \ldots, x_{n-1} \in X$ such that

$$\forall y \in X \exists m < n [d(x_m, y) < 2^{-k}] \, ,$$

and *compact* if it is totally bounded and complete.

The *closure* $\overline{S}$ of a subset S of a metric space X is defined by

$$\overline{S} = \left\{ x \in X \,\middle|\, \forall k \exists y \in S [d(x,y) < 2^{-k}] \right\} .$$

A subset S of a metric space X is *closed* if $\overline{S} = S$. Note that a subset S of a metric space X is closed if and only if $x \in S$ whenever there exists a sequence $(x_n)_{n \geqslant 0}$ of S converging to x. As usual, a metric space X is *separable* if there exists a countable set $D \subset X$ such that $\overline{D} = X$.

A subset S of a metric space X is *located* if it is inhabited and

$$d(x,S) = \inf \{ d(x,y) \,|\, y \in S \}$$

exists for all $x \in X$.

There are various notions of continuity, which, unlike in CLASS, are not always equivalent. Actually, many of the principles we consider exactly embody the jump between these notions.

Definition 10.2.2. Let f be a mapping between metric spaces. Then f is

- *strongly extensional* if $f(x) \# f(y)$ implies $x \# y$;
- *sequentially continuous* if $x_n \to x$ implies $f(x_n) \to f(x)$;
- *discontinuous* if there exist $\delta > 0$ and a sequence $(x_n)_{n \geqslant 0}$ converging to a limit x such that $d(f(x_n), f(x)) \geqslant \delta$ for all n;
- *nondiscontinuous* if $x_n \to x$ and $d(f(x_n), f(x)) \geqslant \delta$ imply $\delta \leqslant 0$;
- *(point-wise) continuous* if

$$\forall x \in X \forall k \exists M_k \forall y \in X \left[d(x,y) < 2^{-M_k} \to d(f(x), f(y)) < 2^{-k} \right] ;$$

- and *uniformly continuous* if

$$\forall k \exists M_k \forall x, y \in X \left[d(x,y) < 2^{-M_k} \to d(f(x), f(y)) < 2^{-k} \right] .$$

A *normed space* is a linear space E equipped with a *norm* $\|\cdot\| : E \to \mathbf{R}$ such that

(N1) $\|x\| = 0 \leftrightarrow x = 0$,
(N2) $\|ax\| = |a| \|x\|$,
(N3) $\|x+y\| \leqslant \|x\| + \|y\|$,

for all $x, y \in E$ and $a \in \mathbf{R}$. Note that a normed space E is a metric space with the metric

$$d(x,y) = \|x - y\| .$$

A *Banach space* is a normed space which is complete with respect to the metric.

A mapping T between linear spaces E and F is *linear* if

1. $T(ax) = aTx$,
2. $T(x+y) = Tx + Ty$

for all $x, y \in E$ and $a \in \mathbf{R}$. A *linear functional* f on a linear space E is a linear mapping from E into $\mathbf{R}$. The *kernel* $\ker(T)$ of a linear mapping T between linear spaces E and F is defined by

$$\ker(T) = \{ x \in E \mid Tx = 0 \} .$$

A linear mapping T between normed spaces E and F is *normable* if

$$\|T\| = \sup \{ \|Tx\| \mid x \in E, \|x\| \leqslant 1 \}$$

exists.

10.3 Omniscience Principles: LPO, WLPO, and LLPO

The *limited principle of omniscience* (LPO, Σ_1-PEM) is of the form

$$\forall \alpha [\alpha \# \mathbf{0} \vee \neg \alpha \# \mathbf{0}] ,$$

and the *weak limited principle of omniscience* (WLPO, Π_1-PEM) is of the form

$$\forall \alpha [\neg \alpha \# \mathbf{0} \vee \neg\neg \alpha \# \mathbf{0}] .$$

They are both universal closures of instances of PEM for Σ_1 and Π_1 formulae, respectively. Alternatively, we can view WLPO as the universal closure of WPEM for Σ_1 formulae.

LPO is a strong principle that enables us to show almost all theorems in (countable) mathematics, and especially in analysis. The following equivalences are due to Mandelkern [56], Ishihara [42], and Diener and Loeb [27].

Proposition 10.3.1. *The following are equivalent.*

1. LPO
2. $\forall x, y \in \mathbf{R} \, (x \# y \vee x = y)$,
3. *Every bounded monotone sequence of real numbers converges.*
4. *Every sequence of a compact metric space has a convergent subsequence.*
5. *Ascoli's theorem.*
6. *Every countable subset S of $\mathbf{R}$ with an upper bound has a supremum,*
7. *Every countable subset of a metric space is located.*
8. *Every bounded linear functional on a separable normed space is normable.*

The following equivalences are due to Mandelkern [55], and van Atten and van Dalen [3].

Proposition 10.3.2. *The following are equivalent.*

1. WLPO
2. $\forall x, y \in \mathbf{R} (\neg x = y \vee x = y)$.
3. *There exists a discontinuous mapping from $\mathbf{N}^{\mathbf{N}}$ into $\mathbf{N}$.*
4. *Every countable subset S of $\mathbf{R}$ with an upper bound has a least upper bound.*

Interestingly enough, not just WLPO has notable equivalences, also its negation does [40].

Proposition 10.3.3. *The following are equivalent.*

1. ¬WLPO
2. *Every mapping from a complete metric space into a metric space is nondiscontinuous.*

The *lesser limited principle of omniscience* (LLPO, Σ_1-DML) is of the form

$$\forall \alpha, \beta \, [\neg(\alpha \# \mathbf{0} \wedge \beta \# \mathbf{0}) \to \neg \alpha \# \mathbf{0} \vee \neg \beta \# \mathbf{0}] \ ,$$

which is a universal closure of an instance of *De Morgan's law* (DML) for Σ_1 formulae A and B.

The following equivalences are due to Mandelkern [54, 55], Ishihara [38], Bridges [18], and Hendtlass [34].

Proposition 10.3.4. *The following are equivalent.*

1. LLPO
2. $\forall x, y \in \mathbf{R}(x \leqslant y \vee y \leqslant x)$.
3. $\forall x, y \in \mathbf{R}\,(xy = 0 \to x = 0 \vee y = 0)$, *that is* $\mathbf{R}$ *is an integral domain.*
4. *Every real number has a binary expansion.*
5. *The intermediate value theorem* (IVT).
6. *Weak Kőnig's Lemma: every infinite, decidable tree has a branch* (WKL).
7. *Every sequence of located closed subsets of a compact metric space with the finite intersection property has an inhabited interior* (CIT).
8. *Every real-valued uniformly continuous function on a compact metric space attains its minimum* (MIN).
9. *The Hahn-Banach theorem: every nonzero bounded linear functional f on a subspace of a separable normed space E, whose kernel is located in E, has an extension g with $\|g\| = \|f\|$.*
10. *Peano's existence theorem for ordinary differential equations.*[4]
11. *Brouwer's Fixed-Point theorem: every uniformly continuous function $f : [0,1]^n \to [0,1]^n$ has a fixed-point.*

It is straightforward to see that

$$\mathrm{LPO} \Rightarrow \mathrm{WLPO} \Rightarrow \mathrm{LLPO} \ .$$

[4] This is for the most direct interpretation of the classical statement. A weaker version of this theorem is equivalent to UCT [28].

10.4 Markov's Principle and Related Principles: MP, WMP, and $\mathrm{MP}^{\vee}$

Markov's principle (MP),[5] which is accepted in RUSS, where it encapsulates the idea of unbounded search, is of the form:

$$\forall \alpha [\neg\neg \alpha \# \mathbf{0} \to \alpha \# \mathbf{0}] \ ,$$

which is equivalent to

$$\forall \alpha [\neg \alpha = \mathbf{0} \to \alpha \# \mathbf{0}] \ .$$

It is a universal closure of instances of the *double negation elimination* for Σ_1 formulae; see [65, 1.11.5] and [67, Section 4.5]. It is known that there are a couple of principles which are weaker than MP. Ishihara [41] studied the weak versions of MP, *weak Markov's principle* (WMP):

$$\forall \alpha [\forall \beta (\neg\neg(\mathbf{0} \# \beta) \vee \neg\neg \beta \# \alpha) \to \alpha \# \mathbf{0}]$$

and *disjunctive Markov's principle* ($\mathrm{MP}^{\vee}$):

$$\forall \alpha, \beta [\neg(\neg \alpha \# \mathbf{0} \wedge \neg \beta \# \mathbf{0}) \to \neg\neg \alpha \# \mathbf{0} \vee \neg\neg \beta \# \mathbf{0}] \ .$$

The principle $\mathrm{MP}^{\vee}$ is a universal closure of DML for Π_1 formulae.

Recently, the following principle, called III_a or Δ_1-PEM, which is even weaker than $\mathrm{MP}^{\vee}$, was found [32] and [50].

$$\forall \alpha, \beta [(\alpha \# \mathbf{0} \leftrightarrow \neg \beta \# \mathbf{0}) \to \alpha \# \mathbf{0} \vee \neg \alpha \# \mathbf{0}] \ .$$

The following are either straightforward, or can be found in the aforementioned [32, 41].

Lemma 10.4.1.

1. $\mathrm{LPO} \Rightarrow \mathrm{MP}$,
2. $\mathrm{LLPO} \Rightarrow \mathrm{MP}^{\vee}$,
3. $\mathrm{WLPO} + \mathrm{MP} \Leftrightarrow \mathrm{LPO}$,
4. $\mathrm{WMP} + \mathrm{MP}^{\vee} \Leftrightarrow \mathrm{MP}$,
5. $\mathrm{MP}^{\vee} \Rightarrow \mathrm{III}_a$.

We can sum up these relationships in the diagram in Figure 10.1.

Proposition 10.4.2.

1. $\mathrm{MP} \Leftrightarrow \forall x, y \in \mathbf{R}(\neg x = y \to x \# y)$,
2. $\mathrm{MP}^{\vee} \Leftrightarrow \forall x, y, z \in \mathbf{R}(\neg x = y \to \neg x = z \vee \neg z = y)$,
3. $\mathrm{WMP} \Leftrightarrow \forall x, y \in \mathbf{R}(\forall z \in \mathbf{R}(\neg x = z \vee \neg z = y) \to x \# y)$.

[5] Note that Markov's principle is also often simply denoted by M.

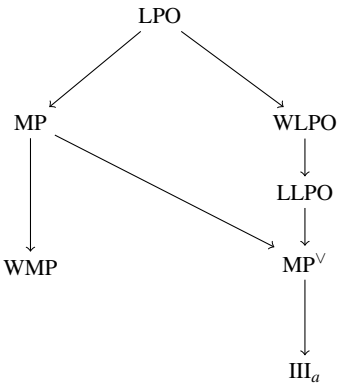


Fig. 10.1 Markov's principle and related principles.

The following proposition is due to Mandelkern [55].

Proposition 10.4.3.

1. $\mathrm{MP}^\vee \Leftrightarrow \forall x,y \in \mathbf{R}\,[\neg x = y \to \{x,y\} \text{ is closed}]$,
2. $\mathrm{WMP} \Leftrightarrow \forall x,y \in \mathbf{R}\,[\{x,y\} \text{ is closed} \to (\neg x = y \to x \# y)]$.

Remark 10.4.4. Mandelkern first considered weak Markov's principle under the names ASP and WLPE in [54] and [56], respectively. Originally, it was stated as "Every pseudo-positive real number is positive," or as a formula:

$$\forall x \in \mathbf{R}\,(\forall y \in \mathbf{R}(\neg\neg(0 < y) \vee (\neg\neg(y < x))) \to x > 0)\ .$$

Kohlenbach [48] showed that WMP is underivable in $\mathbf{E}\text{-}\mathbf{HA}^\omega + \mathrm{AC} + \mathrm{CA}^\omega_\neg$; see also [49, 7.1].

The following equivalences are due to Bridges and Ishihara [20].

Proposition 10.4.5. *The following are equivalent.*

1. MP
2. *Every mapping between metric spaces is strongly extensional.*
3. *Every linear mapping between normed spaces is strongly extensional.*

A subtle change happens, when we change our focus to complete spaces, as the following theorem due to Ishihara [40] shows.

Proposition 10.4.6. *The following are equivalent.*

1. WMP
2. *Every mapping from a complete metric space into a metric space is strongly extensional.*

3. *Every nondiscontinuous mapping from a complete metric space into a metric space is sequentially continuous.*

It is worth pointing out that in the case of linear maps there is no need to assume any form of Markov's principle, as shown in the result by Bridges and Ishihara [20].

Proposition 10.4.7. *Every linear mapping from a Banach space into a normed space is strongly extensional.*

10.5 A Continuity Principle, the Fan Theorem, and Church's Thesis

The following principle is assumed in INT.

The principle of *weak continuity for numbers* (WC-N):

$$\forall\alpha\exists nA(\alpha,n)\to\forall\alpha\exists mn\forall\beta\in\overline{\alpha}(m)A(\beta,n)\ .$$

WC-N has some consequences which put its acceptance at odds with omniscience principles such as LLPO [67, 40].[6]

Proposition 10.5.1. *Assume* WC-N. *Then*

1. ¬LLPO
2. ¬WKL
3. WMP

Brouwer's name is also linked with bar induction and a consequence thereof: the fan theorem, which formalises the idea of $\{0,1\}^{\mathbf{N}}$ being cover compact.

The (full) fan theorem (FAN):

$$\forall\alpha\in\{0,1\}^{\mathbf{N}}\exists nA(\overline{\alpha}(n))\to\exists n\forall\alpha\in\{0,1\}^{\mathbf{N}}\exists k\leqslant nA(\overline{\alpha}(k))\ .$$

Notice that Brouwer assumed the principle of continuous choice, which is even stronger than WC-N, and which meant that the complexity of the predicate A in the fan theorem did not matter. In the absence of continuous choice we do, however, have to distinguish between different versions of the fan theorem. The weakest is the one for decidable A and will be denoted by FAN_{Δ}. There are also stronger versions of the fan theorem in common use: FAN_c and $\mathrm{FAN}_{\Pi^1_0}$. These are the fan theorems for c-bars, and for Π_1-bars. Here a predicate A on $\{0,1\}^*$ is a c-predicate[7] if there exists a decidable B such that

[6] Note that in [67] LLPO is called SEP.

[7] In [8] Berger and Bridges give the following reason for using the prefix c in defining the analogous notion for sets: "The letter c in the expression c-set should indicate that this notion of complexity is related to continuity".

$$A(u) \leftrightarrow \forall w \in \{0,1\}^* \, B(u * w) \; .$$

In this context, a bar A is a Π_1-bar if there exists a decidable predicate S on $\{0,1\}^* \times \mathbf{N}$ such that

1. $A(u) \leftrightarrow \forall n \in \mathbf{N}(S(u,n))$
2. If $S(u,n)$, then for any $w \in \{0,1\}^*$ also $S(u * w, n)$.

Notice that the second condition here is not implied by the name, but included for technical reasons.

We have (UCT will be introduced below)

$$\mathrm{FAN}_{\Pi_0^1} \Rightarrow \mathrm{UCT} \Rightarrow \mathrm{FAN}_c \Rightarrow \mathrm{FAN}_\Delta \; .$$

All of these implications are straightforward, apart from, possibly, $\mathrm{FAN}_{\Pi_0^1} \Rightarrow \mathrm{UCT}$, which is proven in [27, Theorem 16].

Viewed through classical eyes the fan theorem(s) can be seen as the contrapositive of the aforementioned WKL. However, viewed constructively the latter is stronger. Since LLPO is equivalent to WKL, the following result is not surprising.

Proposition 10.5.2. $\mathrm{LLPO} \vdash \mathrm{FAN}_c$.[8]

A direct proof can be found in [7]. Since the (full) fan theorem holds in INT, yet WKL fails there, there is no hope of proving the converse of this.

The following equivalences are due to Julian and Richman [47], Berger and Ishihara [10], Diener [23], and Berger and Schuster [12].

Proposition 10.5.3. *The following are equivalent.*

1. FAN_Δ
2. *Every infinite, decidable tree with at most one path has an infinite path* (WKL!).[9]
3. *Every uniformly continuous, real-valued function on a compact metric space with at most one minimum point has a minimum point* (MIN!).[10]
4. *Every positive-valued function on* $[0,1]$ *has a positive infimum* (POS).
5. *(A version of) the Heine Borel Theorem: every countable cover of* $[0,1]$ *with open intervals has a finite subcover.*
6. *Dini's theorem: if* $(f_n)_{n \geqslant 0} : [0,1] \to \mathbf{R}$ *is an increasing sequence of uniformly continuous functions converging point-wise to a uniformly continuous* $f : [0,1] \to \mathbf{R}$*, then the convergence is uniform.*

The following equivalences are due to Berger [6], and Berger and Bridges [8, 9].

Proposition 10.5.4. *The following are equivalent.*

[8] With a lot of technical effort one can also prove the stronger statement that $\mathrm{LLPO} \vdash \mathrm{UCT}$, as shown in [25]. Since FAN_Δ fails in RUSS, but $\mathrm{MP}^\vee$ holds there, we cannot hope to weaken the antecedent to $\mathrm{MP}^\vee$ or III_a

[9] A tree T *has at most one path* if $\alpha \neq \beta$ implies $\overline{\alpha}(n) \notin T \vee \overline{\beta}(n) \notin T$ for some n.

[10] A real-valued function f on a metric space X *has at most one minimum point* if $x \# y$ implies $f(z) < f(x)$ or $f(z) < f(y)$ for some $z \in X$.

1. FAN_c
2. *Every (point-wise) continuous mapping* $\{0,1\}^{\mathbf{N}} \to \mathbf{N}$ *is uniformly continuous.*
3. *The Anti-Specker property: every sequence* $(x_n)_{n\geqslant 0}$ *of reals that is eventually bounded away from every point in* $[0,1]$ *is eventually bounded away from the entire set* $[0,1]$ (AS).

Notice that it is, to our best knowledge, unknown whether FAN_c is actually equivalent to the stronger statement that every continuous mapping from a compact metric space into a metric space is uniformly continuous (UCT).

We do however know the following [19].

Proposition 10.5.5. *The following are equivalent.*

1. UCT
2. *Every (point-wise) continuous mapping* $[0,1] \to \mathbf{R}$ *is uniformly continuous.*
3. *Every (point-wise) continuous mapping* $[0,1] \to \mathbf{R}$ *is bounded.*

There are some equivalences to $\mathrm{FAN}_{\Pi^1_0}$ [8, 24], but they seem to be somewhat less natural than the ones to FAN_Δ, FAN_c, and UCT.

Related to the topic of the fan theorems, one can also consider a principle weaker than FAN_Δ which is obtained by weakening the conclusion of FAN_Δ from requiring "uniform blocking of sequences" to requiring that the ratio of sequences of a given length which are "blocked" by the bar over those that are not approaches 1. This principle is known as *weak weak Kőnig's lemma* (WWKL), and some equivalences to it have been investigated by Nemoto [58], and Diener and Hedin [26]. The principle WWKL was first considered by researchers working in "Simpson-style" reverse mathematics, who also coined the name. More recently, it has also been investigated with respect to the Weihrauch lattice [17].

In the same way, and totally disregarding philosophical aspects, that we can view INT as BISH plus assuming continuous choice and bar induction, we can view RUSS as BISH plus the assumption of some form of Church's thesis (CT), which states that each sequence of natural numbers is computable. It can be expressed by an axiom as

$$\forall\alpha\exists k\forall n\exists m(T(k,n,m)\wedge U(m)=\alpha(n))$$

where T and U are Kleene's T-predicate and the result-extracting function, respectively. Combining this with a restricted form of AC:

$$\forall n\exists m B(n,m)\to\exists\alpha\forall n B(n,\alpha(n))\ ,$$

we have the arithmetical form of Church's thesis (CT_0):

$$\forall n\exists m B(n,m)\to\exists k\forall n\exists m[T(k,n,m)\wedge B(n,U(m))],$$

which has an extended form (ECT_0):

$$\forall n[A(n)\to\exists m B(n,m)]\to\exists k\forall n[A(n)\to\exists m(T(k,n,m)\wedge B(n,U(m)))]$$

where A is almost negative.[11]

Proposition 10.5.6. *Assume* CT_0. *Then*

1. $\neg$LLPO
2. WMP
3. $\neg$WC-N
4. $\neg\mathrm{FAN}_\Delta$
5. *There exists a bounded monotone sequence* $(q_n)_{n\geqslant 0}$ *of rational numbers such that for all* $x \in \mathbf{R}$
$$\exists mk \forall n \geqslant m(2^{-k} < |x - q_n|) \ .$$
6. *There exists a continuous mapping from* $\{0,1\}^{\mathbf{N}}$ *into* $\mathbf{N}$ *which is not uniformly continuous.*

Proposition 10.5.7. *Assume* ECT_0. *Then* III_a *implies* $\mathrm{MP}^\vee$.

10.6 The Boundedness Principle: BD-N

Together with the aforementioned WMP there is one more principle that takes a special place in constructive reverse mathematics, in that it holds in the three major varieties of constructive mathematics CLASS, INT, and RUSS, but is not accepted by practitioners of BISH. Before we can state this principle, we need a definition.

Definition 10.6.1. A subset S of $\mathbf{N}$ is *pseudobounded* if $\lim_{n\to\infty} s_n/n = 0$ for each sequence $(s_n)_{n\geqslant 0}$ in S.

In [46, Lemma 3], it was shown that a set A of natural numbers is pseudobounded if and only if for each sequence $(a_n)_{n\geqslant 0}$ in A, $a_n < n$ for all sufficiently large n. A few further characterisations can be found in [61].

Every bounded subset is trivially pseudobounded and, conversely, every inhabited, *decidable*, and pseudobounded subset of $\mathbf{N}$ is easily seen to be bounded. However, in the absence of decidability this is not guaranteed anymore, which motivates the boundedness principle BD-N:

Every countable pseudobounded subset is bounded.

The following equivalences are due to Ishihara [39] and Ishihara-Yoshida [46], Bridges-Ishihara [21], and Ishihara [44].

Proposition 10.6.2. *The following are equivalent.*

1. BD-N
2. *Every sequentially continuous mapping from a separable metric space into a metric space is point-wise continuous.*

[11] A formula is called *almost negative* if it has no $\vee$ and has $\exists$ only immediately preceding prime formulae [4, p.155].

3. *The space of test functions $\mathscr{D}(\mathbf{R})$ is (sequentially) complete.*[12]
4. *(A version of) the open mapping theorem in functional analysis: If T is a nonzero bounded linear mapping of a separable Hilbert space H into itself such that T^* exists and* $\mathrm{ran}(T)$ *is complete, then T is open.*
5. *Every one-one self-adjoint sequentially continuous linear mapping from a Hilbert space onto itself is bounded.*
6. *If $(T_m)_{m\geqslant 1}$ is a sequence of bounded linear mappings from a separable Banach space E into a normed space F such that the set*

$$\{\, T_m x \,|\, m \in \mathbf{N} \,\}$$

is bounded for each $x \in E$, then $(T_m)_{m\geqslant 1}$ is equicontinuous.

Under the assumption of BD-N the hierarchy of fan theorems introduced in the previous section somewhat collapses, as shown in [23, Chapter 4].

Proposition 10.6.3. $\text{BD-N} \vdash \text{FAN}_c \to \text{FAN}_{\Pi^1_0}$

As mentioned before, BD-N is valid in both RUSS and INT. More precisely we have the following [40].

Proposition 10.6.4. $\text{ECT}_0 + \text{MP} \vdash \text{BD-N}$ *and* $\text{WC-N} \vdash \text{BD-N}$

It is also trivial to see that $\text{LPO} \vdash \text{BD-N}$, whence the latter holds in CLASS.

Remark 10.6.5. Lietz and Streicher [52] showed that BD-N is underivable in $\mathbf{E}\text{-}\mathbf{HA}^\omega + \text{AC}$. Using topological models Lubarsky [53] has shown that BD-N is underivable even in the presence of $\text{IZF} + \text{DC}$.

We would like to finish this section by pointing out that BD-N is the last puzzle piece to a clean, axiomatic proof of the Kreisel-Lacombe-Shoenfield theorem (KLST) that all real-valued functions on a complete, separable metric space are point-wise continuous.[13]

Proposition 10.6.6. *The following are equivalent.*

1. $\neg\text{WLPO} + \text{WMP} + \text{BD-N}$
2. KLST

Proof. Let $f : X \to \mathbf{R}$ be a real-valued function on a complete and separable metric space X. Proposition 10.3.3 shows that $\neg$WLPO implies that f is nondiscontinuous.

[12] The space of all test functions $\mathscr{D}(\mathbf{R})$ is the space of all infinitely differentiable functions $f : \mathbf{R} \to \mathbf{R}$ with compact support together with a locally convex structure defined by the seminorms

$$p_{\alpha,\beta}(f) := \sup_n \max_{\ell\leqslant\beta(n)} \sup_{|x|>n} 2^{\alpha(n)} |f^{(\ell)}(x)| \qquad (\alpha, \beta \in \mathbf{N} \to \mathbf{N}) \ .$$

[13] Of course, this is talking about a constructive interpretation of KLST. However, using an appropriate realizability interpretation (or interpreted informally), the proof of the forward direction will yield a proof in classical computability theory.

The assumption of WMP now guarantees that f is therefore sequentially continuous, by Proposition 10.4.6. Finally, BD-N implies that f is point-wise continuous by Proposition 10.6.2.

The proof of the converse can be found in [40].

Thus, combining the previous theorem with the results in Sections 10.4 and 10.5, we also have the following.

Corollary 10.6.7. $\mathrm{ECT}_0 + \mathrm{MP} \vdash \mathrm{KLST}$, *and* $\mathrm{WC\text{-}N} \vdash \mathrm{KLST}$.

10.7 Relationships Between Principles

The diagram in Figure 10.2 should give a good overview over the various principles discussed, and in particular their relationships among each other. Arrows mean implication, and an arrow from A to C and labelled by B means that A together with B implies C.

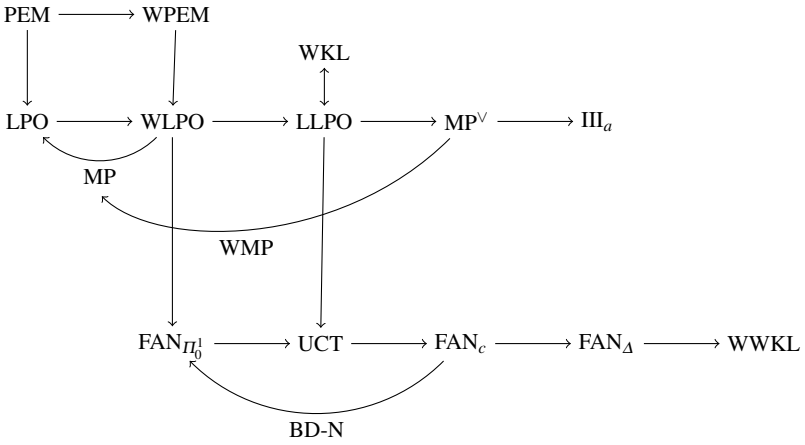


Fig. 10.2 Relationships between principles.

10.8 Separation Techniques

The easiest and most convenient way to see that principle A does not imply principle B, or, more generally, that theorem T is *not provable* in BISH is to show that theorem T is *false* in CLASS, INT, or RUSS. These three varieties are all well understood so they immediately show many separations between the principles we have discussed

so far. For example MP does not imply FAN_Δ, since the first one is true in RUSS, whereas the second one is false there.

To obtain subtler separations between principles, such as the one between LPO and WLPO, one needs more refined methods. Various approaches have been used. Before describing some of these methods we should mention that, since BISH is not formalised (in the same way that CLASS is not formalised), it is somewhat problematic to speak of "models of BISH". Of course, various formal systems designed to capture BISH have been proposed over the years, such as Myhill's constructive set theory [57], Aczel's Constructive ZF-style set theory CZF [1], or Veldman's BIM [68], and if one chooses one of these formalisations, one can then formally show the separation of principles over these formal systems. The, in terms of foundations, non-committal nature of BISH might seem like a big problem for constructive reverse mathematics, however, in practice this is not really an issue. Topological models, for example, are sound for intuitionistic ZF set theory (IZF) [35, Theorem 16], which is a much stronger set theory than most constructivists need for their mathematics and much stronger than the systems mentioned above. Thus a separation shown with the help of topological models is enough evidence to treat the respective principles as separate in constructive reverse mathematics.

There is one issue that is, indeed, problematic in this context and that is, again, the issue of countable/dependent choice. Topological and realizability models do not always validate these principles, so whether they hold or not needs to be checked separately. The specific topological models cited below all validate countable/dependent choice, which makes the decision of whether one accepts these principles or not independent of the separation results.

Topological models are a natural setting to interpret formalised intuitionistic theories. By "intuitionistic" we mean theories using intuitionistic logic; it is worth noting though, that topological models also have a distinct intuitionistic flavour à la Brouwer. For example they all validate the full, unrestricted, fan theorem [30, Theorem 3.2]. Topological models are actually just a special case of Heyting-valued models which have a long history starting with several publications around 1970 [62, 63, 33, 30]. Topological models have been used to give models that do not satisfy BD-N [53], separate LPO, WLPO, LLPO, MP, WMP, PEM, WPEM, and various variations of these [36].

An approach which is very different from topological models is that of realizability models [59]. In these we extend the logical language by allowing witnesses of statements to be attached to statements. This is, in particular, interesting in a constructive context, where we want to attach computable objects ("realizers") that describe the computational content of a formula. There are many different ways to fix details of how the logical connectives and quantifiers are handled, leading to very different interpretations. Most realizability models have a distinct recursive flavour and so we can say that, vaguely speaking, realizability models are to RUSS what topological models are to INT. We will not go into any details and only point the reader to [4, Chapter 7] and [59] for further details. As mentioned in the respective section, realizability models have been used to show that BD-N is not derivable in intuitionistic logic [52].

Finally, one can also use various proof interpretations [49], which, vaguely speaking, can also be seen as types of realizabilities, to separate principles. As already mentioned, this was used to first show that WMP is not derivable [48], which implies that WLPO does not imply MP.

Acknowledgements The second author thanks the Japan Society for the Promotion of Science (JSPS), Core-to-Core Program (A. Advanced Research Networks) for supporting the research.

References

1. Aczel, P., Rathjen, M.: Notes on constructive set theory. Tech. Rep. 40, Institut Mittag-Leffler, The Royal Swedish Academy of Sciences (2001)
2. Akama, Y., Berardi, S., Hayashi, S., Kohlenbach, U.: An arithmetical hierarchy of the law of excluded middle and related principles. In: Proceedings of the 19th Annual IEEE Symposium on Logic in Computer Science (LICS 2004), pp. 192–201. IEEE Computer Society Press (2004)
3. van Atten, M., van Dalen, D.: Arguments for the continuity principle. Bulletin of Symbolic Logic **8**(3), 329–347 (2002)
4. Beeson, M.J.: Foundations of Constructive Mathematics. *A Series of Modern Surveys in Mathematics*. Springer, Berlin (1985)
5. Berardi, S.: Some intuitionistic equivalents of classical principles for degree 2 formulas. Annals of Pure and Applied Logic **139**(1), 185–200 (2006)
6. Berger, J.: The fan theorem and uniform continuity. In: S.B. Cooper, B. Löwe, L. Torenvliet (eds.) New Computational Paradigms, *Lecture Notes in Computer Science*, vol. 3526, pp. 18–22. Springer-Verlag, Berlin, Heidelberg (2005), Proceedings of the First International Conference on Computability in Europe, CiE 2005
7. Berger, J.: Weak König's lemma implies the fan theorem for c-bars. In: Computation and Logic in the Real World, pp. 86–88. University of Siena (2007), CiE 2007, Quaderni del Dipartimento di Scienze Matematiche e Informatiche "Roberto Magari"
8. Berger, J., Bridges, D.S.: A bizarre property equivalent to the Π^0_1–fan theorem. Logic Journal of the IGPL **14**, 867–871 (2006)
9. Berger, J., Bridges, D.S.: A fan-theoretic equivalent of the antithesis of Specker's theorem. Indagationes Mathematicae **18**(2), 195–202 (2007)
10. Berger, J., Ishihara, H.: Brouwer's fan theorem and unique existence in constructive analysis. Mathematical Logic Quarterly **51**(4), 360–364 (2005)
11. Berger, J., Ishihara, H., Kihara, T., Nemoto, T.: The binary expansion and the intermediate value theorem in constructive reverse mathematics. Archive for Mathematical Logic **58**(1-2), 203–217 (2019)
12. Berger, J., Schuster, P.: Classifying Dini's theorem. Notre Dame Journal of Formal Logic **47**(2), 253–262 (2006)
13. Bishop, E.: Foundations of Constructive Analysis. McGraw-Hill, New York (1967)
14. Bishop, E., Bridges, D.S.: Constructive Analysis, *Grundlehren der Mathematischen Wissenschaften*, vol. 279. Springer, Berlin (1985)
15. Brattka, V., Gherardi, G.: Effective choice and boundedness principles in computable analysis. The Bulletin of Symbolic Logic **17**(1), 73–117 (2011)
16. Brattka, V., Gherardi, G.: Weihrauch degrees, omniscience principles and weak computability. The Journal of Symbolic Logic **76**(1), 143–176 (2011)
17. Brattka, V., Gherardi, G., Hölzl, R., Pauly, A.: The Vitali covering theorem in the Weihrauch lattice. In: A. Day, M. Fellows, N. Greenberg, B. Khoussainov, A. Melnikov, F. Rosamond (eds.) Computability and Complexity: Essays Dedicated to Rodney G. Downey on the Occasion of His 60th Birthday, *Lecture Notes in Computer Science*, vol. 10010, pp. 188–200. Springer, Cham (2017)

18. Bridges, D.S.: Constructive solutions of ordinary differential equations. In: U. Berger, H. Diener, P. Schuster, M. Seisenberger (eds.) Logic, Construction, Computation, *Ontos Mathematical Logic*, vol. 3, pp. 67–77. de Gruyter (2012)
19. Bridges, D.S., Diener, H.: The pseudocompactness of $[0,1]$ is equivalent to the uniform continuity theorem. The Journal of Symbolic Logic **72**(4), 1379–1384 (2007)
20. Bridges, D.S., Ishihara, H.: Linear mappings are fairly well-behaved. Archiv der Mathematik **54**(6), 558–562 (1990)
21. Bridges, D.S., Ishihara, H.: A definite constructive open mapping theorem? Mathematical Logic Quarterly **44**(4), 545–552 (1998)
22. Bridges, D.S., Richman, F.: Varieties of Constructive Mathematics, *London Mathematical Society Lecture Note Series*, vol. 97. Cambridge University Press, Cambridge (1987)
23. Diener, H.: Compactness under constructive scrutiny. Ph.D. thesis. University of Canterbury, Christchurch, New Zealand (2008)
24. Diener, H.: Reclassifying the antithesis of Specker's theorem. Arch. Math. Logic **51**(7-8), 687–693 (2012)
25. Diener, H.: Weak König's lemma implies the uniform continuity theorem. Computability **2**(1), 9–13 (2013)
26. Diener, H., Hedin, A.: The Vitali covering theorem in constructive mathematics. Journal of Logic and Analysis **4**, Paper 7, 22 (2012)
27. Diener, H., Loeb, I.: Sequences of real functions on $[0,1]$ in constructive reverse mathematics. Annals of Pure and Applied Logic **157**(1), 1–62 (2009)
28. Diener, H., Loeb, I.: Constructive reverse investigations into differential equations. Journal of Logic and Analysis **3**, Paper 8, 26 (2011)
29. Dummet, M.: Elements of Intuitionism, 2nd edn. Oxford University Press (2000)
30. Fourman, M., Hyland, J.M.E.: Sheaf models for analysis. In: M. Fourman, C. Mulvey, D. Scott (eds.) Applications of Sheaves, *Lecture Notes in Mathematics*, vol. 753, pp. 280–301. Springer, Berlin/Heidelberg (1979)
31. Friedman, H.: Some systems of second order arithmetic and their use. In: Proceedings of the International Congress of Mathematicians, Vancouver, vol. 1, pp. 235–242 (1974)
32. Fujiwara, M., Ishihara, H., Nemoto, T.: Some principles weaker than Markov's principle. Archive for Mathematical Logic **54**(7-8), 861–870 (2015)
33. Grayson, R.J.: Heyting-valued semantics. In: G. Lolli, G. Longo, A. Marcja (eds.) Logic Colloquium '82, *Studies in Logic and the Foundations of Mathematics*, vol. 112, pp. 181–208. Elsevier (1984)
34. Hendtlass, M.: Fixed point theorems in constructive mathematics. Journal of Logic and Analysis **4**, Paper 10, 20 (2012)
35. Hendtlass, M.: Constructing fixed points and economic equilibria. Ph.D. thesis. Department of Pure Mathematics, University of Leeds, Leeds, UK (2013)
36. Hendtlass, M., Lubarsky, R.: Separating fragments of WLEM, LPO, and MP. Journal of Symbolic Logic **81**(4) (2016)
37. Heyting, A.: Intuitionism: An Introduction, 3rd edn. North-Holland (1971)
38. Ishihara, H.: An omniscience principle, the König lemma and the Hahn-Banach theorem. Zeitschrift für Mathematische Logik und Grundlagen der Mathematik **36**, 237–240 (1990)
39. Ishihara, H.: Continuity and nondiscontinuity in constructive mathematics. The Journal of Symbolic Logic **56**(4), 1349–1354 (1991)
40. Ishihara, H.: Continuity properties in constructive mathematics. The Journal of Symbolic Logic **57**(2), 557–565 (1992)
41. Ishihara, H.: Markov's principle, Church's thesis and Lindelöf's theorem. Indagationes Mathematicae **4**, 321–325 (1993)
42. Ishihara, H.: Sequential continuity of linear mappings in constructive mathematics. Journal of Universal Computer Science **3**, 1250–1254 (1997)
43. Ishihara, H.: Constructive reverse mathematics: compactness properties. In: L. Crosilla, P. Schuster (eds.) From Sets and Types to Topology and Analysis: Towards Practicable Foundations for Constructive Mathematics, pp. 245–267. Oxford University Press (2005)

44. Ishihara, H.: The uniform boundedness theorem and a boundedness principle. Annals of Pure and Applied Logic **163**(8), 1057–1061 (2012)
45. Ishihara, H.: On Brouwer's continuity principle. Indagationes Mathematicae **29**(6), 1511–1524 (2018)
46. Ishihara, H., Yoshida, S.: A constructive look at the completeness of the space $\mathscr{D}(\mathbf{R})$. Journal of Symbolic Logic **67**(4), 1511—1519 (2002)
47. Julian, W., Richman, F.: A uniformly continuous function on $[0,1]$ that is everywhere different from its infimum. Pacific Journal of Mathematics **111**(2), 333–340 (1984)
48. Kohlenbach, U.: On weak Markov's principle. Mathematical Logic Quarterly **48**(Suppl. 1), 59–65 (2002)
49. Kohlenbach, U.: Applied Proof Theory: Proof Interpretations and Their Use in Mathematics. Springer, Berlin (2008)
50. Kohlenbach, U.: On the disjunctive Markov principle. Studia Logica **103**(6), 1313–1317 (2015)
51. Kushner, B.A.: Lectures on Constructive Mathematical Analysis, *Translations of Mathematical Monographs*, vol. 60. American Mathematical Society, Providence (1984)
52. Lietz, P., Streicher, T.: Realizability models refuting Ishihara's boundedness principle. Annals of Pure and Applied Logic **163**(12), 1803–1807 (2012)
53. Lubarsky, R.S.: On the failure of BD-$\mathbb{N}$ and BD, and an application to the anti-Specker property. J. Symbolic Logic **78**(1), 39–56 (2013)
54. Mandelkern, M.: Constructive continuity. Memoirs of the American Mathematical Society **42**(277), v+117 (1983)
55. Mandelkern, M.: Constructively complete finite sets. Zeitschrift für Mathematische Logik und Grundlagen der Mathematik **34**(2), 97–103 (1988)
56. Mandelkern, M.: Limited omniscience and the Bolzano-Weierstrass principle. Bulletin of the London Mathematical Society **20**, 319–320 (1988)
57. Myhill, J.: Constructive set theory. The Journal of Symbolic Logic **40**(3), 347–382 (1975)
58. Nemoto, T.: Weak weak König's lemma in constructive reverse mathematics. In: Proceedings of the 10th Asian Logic Conference (ALC 10), pp. 263—270 (2008), Kobe University in Kobe, Hyogo, Japan, September 1–6, 2008
59. van Oosten, J.: Realizability: An Introduction to Its Categorical Side, *Studies in Logic and the Foundations of Mathematics*, vol. 152. Elsevier B. V., Amsterdam (2008)
60. Richman, F.: Constructive mathematics without choice. In: P. Schuster, U. Berger, H. Osswald (eds.) Reuniting the Antipodes: Constructive and Nonstandard Views of the Continuum, *Synthese Library*, vol. 306. Kluwer A. P. (2001)
61. Richman, F.: Intuitionistic notions of boundedness in $\mathbb{N}$. Mathematical Logic Quarterly **55**(1), 31–36 (2009)
62. Scott, D.: Extending the topological interpretation to intuitionistic analysis. Compositio Mathematica **20**, 194–210 (1968)
63. Scott, D.: Extending the topological interpretation to intuitionistic analysis. II. In: A. Kino, J. Myhill, R.E. Vesley (eds.) Intuitionism and Proof Theory (Proc. Conf., Buffalo, N.Y., 1968), *Studies in Logic and the Foundations of Mathematics*, vol. 60, pp. 235–255. North-Holland, Amsterdam (1970)
64. Simpson, S.G.: Subsystems of Second Order Arithmetic. *Perspectives in Mathematical Logic*. Springer, Berlin (1999)
65. Troelstra, A. (ed.): Metamathematical Investigation of Intuitionistic Arithmetic and Analysis, *Lecture Notes in Mathematics*, vol. 344. Springer, Berlin (1973)
66. Troelstra, A., van Dalen, D.: Constructivism in Mathematics, Volume 1, *Studies in Logic and the Foundations of Mathematics*, vol. 121. North-Holland, Amsterdam (1988)
67. Troelstra, A., van Dalen, D.: Constructivism in Mathematics, Volume 2, *Studies in Logic and the Foundations of Mathematics*, vol. 123. North-Holland, Amsterdam (1988)
68. Veldman, W.: Brouwer's fan theorem as an axiom and as a contrast to Kleene's alternative. Archive for Mathematical Logic **53**(5), 621–693 (2014)

Chapter 11
Weihrauch Complexity in Computable Analysis

Vasco Brattka, Guido Gherardi and Arno Pauly

Abstract We provide a self-contained introduction to Weihrauch complexity and its applications to computable analysis. This includes a survey on some classification results and a discussion of its relation to other approaches.

11.1 The Algebra of Problems

The Weihrauch lattice offers a framework to classify the uniform computational content of problems and theorems from analysis and other areas of mathematics. This framework can be seen as an attempt to create a calculus of mathematical problems, very much in spirit of Kolmogorov's interpretation of intuitionistic logic [71].

We express mathematical problems with the help of partial multi-valued functions $f :\subseteq X \rightrightarrows Y$, which are just relations $f \subseteq X \times Y$. It has turned out to be fruitful for our approach to think of these relations as input-output-oriented multi-valued functions $f :\subseteq X \rightrightarrows Y$. We consider $\mathrm{dom}(f) = \{x \in X : f(x) \neq \emptyset\}$ as the set of admissible instances x of the problem f, and we consider the corresponding set of function values $f(x) \subseteq Y$ as the set of possible results. In the case of single-valued f we identify $f(x)$ with the corresponding singleton. An example of a mathematical

Vasco Brattka
Faculty of Computer Science, Universität der Bundeswehr München, Germany and
Department of Mathematics & Applied Mathematics, University of Cape Town, South Africa
e-mail: Vasco.Brattka@cca-net.de

Guido Gherardi
Dipartimento di Filosofia e Comunicazione, Università di Bologna, Italy
e-mail: Guido.Gherardi@unibo.it

Arno Pauly
Department of Computer Science, Swansea University, United Kingdom
e-mail: a.m.pauly@swansea.ac.uk

V. Brattka, P. Hertling (eds.), *Handbook of Computability and Complexity in Analysis*,
Theory and Applications of Computability, https://doi.org/10.1007/978-3-030-59234-9_11

problem that the reader can have in mind as a prototypical case is the zero problem. Obviously, many problems in mathematics can be expressed in terms of solutions of equations of the type $f(x) = 0$ with a continuous $f : X \to \mathbb{R}$. We formalize this problem.

Example 11.1.1 (Zero problem). Let X be a topological space and let $\mathscr{C}(X)$ denote the set of continuous $f : X \to \mathbb{R}$. The *zero problem* $\mathsf{Z}_X :\subseteq \mathscr{C}(X) \rightrightarrows X, f \mapsto f^{-1}\{0\}$ is the problem to find a solution $x \in X$ of an equation of the type $f(x) = 0$, given a continuous function $f : X \to \mathbb{R}$. The set $\mathrm{dom}(\mathsf{Z}_X)$ of admissible instances of this problem is the set of all continuous functions f with a non-empty zero set $f^{-1}\{0\}$. The set $\mathsf{Z}_X(f) = f^{-1}\{0\}$ of solutions is the set of all zeros of f.

Mathematical problems can be combined in various natural ways to obtain new problems. The following definition lists a number of typical algebraic operations that we are going to use. By $X \sqcup Y := (\{0\} \times X) \cup (\{1\} \times Y)$ we denote the *disjoint union*. By $X^* := \bigcup_{i=0}^{\infty}(\{i\} \times X^i)$ we denote the *set of words* over X, where $X^i := \bigtimes_{j=1}^{i} X$ stands for the i–fold *Cartesian product* of X with itself with $X^0 := \{()\}$. Here $()$ stands for the empty tuple or word. By $\overline{X} := X \cup \{\bot\}$ we denote the *completion* of X, where $\bot \notin X$. We use the set of *natural numbers* $\mathbb{N} = \{0, 1, 2, ...\}$.

Definition 11.1.2 (Algebraic operations). Let $f :\subseteq X \rightrightarrows Y$, $g :\subseteq Z \rightrightarrows W$ and $h :\subseteq Y \rightrightarrows Z$ be multi-valued functions. We define the following operations (for exactly those inputs given by the specified domains):

1. $h \circ f :\subseteq X \rightrightarrows Z$, $(h \circ f)(x) := \{z \in Z : (\exists y \in f(x))\ z \in h(y)\}$ and $\mathrm{dom}(h \circ f) := \{x \in \mathrm{dom}(f) : f(x) \subseteq \mathrm{dom}(h)\}$ (composition)
2. $f \times g :\subseteq X \times Z \rightrightarrows Y \times W, (f \times g)(x,z) := f(x) \times g(z)$ and $\mathrm{dom}(f \times g) := \mathrm{dom}(f) \times \mathrm{dom}(g)$ (product)
3. $f \sqcup g :\subseteq X \sqcup Z \rightrightarrows Y \sqcup W$, $(f \sqcup g)(0,x) := \{0\} \times f(x)$, $(f \sqcup g)(1,z) := \{1\} \times g(z)$ and $\mathrm{dom}(f \sqcup g) := \mathrm{dom}(f) \sqcup \mathrm{dom}(g)$ (coproduct)
4. $f \boxplus g :\subseteq X \sqcup Z \rightrightarrows \overline{Y} \times \overline{W}$, $(f \boxplus g)(0,x) := f(x) \times \overline{W}$, $(f \boxplus g)(1,z) := \overline{Y} \times g(z)$ and $\mathrm{dom}(f \boxplus g) := \mathrm{dom}(f) \sqcup \mathrm{dom}(g)$ (box sum)
5. $f \sqcap g :\subseteq X \times Z \rightrightarrows Y \sqcup W, (f \sqcap g)(x,z) := f(x) \sqcup g(z)$ and $\mathrm{dom}(f \sqcap g) := \mathrm{dom}(f) \times \mathrm{dom}(g)$ (meet)
6. $f + g :\subseteq X \times Z \rightrightarrows \overline{Y} \times \overline{W}, (f+g)(x,z) := (f(x) \times \overline{W}) \cup (\overline{Y} \times g(z))$ and $\mathrm{dom}(f+g) := \mathrm{dom}(f) \times \mathrm{dom}(g)$ (sum)
7. $f^* :\subseteq X^* \rightrightarrows Y^*, f^*(i,x) := \{i\} \times f^i(x)$ and $\mathrm{dom}(f^*) := \mathrm{dom}(f)^*$ (finite parallelization)
8. $\widehat{f} :\subseteq X^{\mathbb{N}} \rightrightarrows Y^{\mathbb{N}}, \widehat{f}(x_n)_n := \bigtimes_{i \in \mathbb{N}} f(x_i)$ and $\mathrm{dom}(\widehat{f}) := \mathrm{dom}(f)^{\mathbb{N}}$ (parallelization)

Here $f^i := \bigtimes_{j=1}^{i} f$ denotes the i–fold product of f with itself, where $f^0 = \mathrm{id}_{X^0}$. It is important to point out that the appropriate definition of the domain of $h \circ f$ is crucial. If $x \in \mathrm{dom}(h \circ f)$, then we require that all possible results $y \in f(x)$ of f

upon input of x are supported by h, i.e., $f(x) \subseteq \mathrm{dom}(h)$. This definition of composition corresponds to our understanding of multi-valued functions as computational problems.[1] We often write for short hf for the composition $h \circ f$.

The reader might notice some relations between the resource-oriented interpretation of linear logic and the way we combine mathematical problems (see Section 11.9.1). Indeed, the following intuitive interpretation of some of our algebraic operations is useful:

1. The composition $h \circ f$ applies both problems consecutively, first f and then h.
2. The product $f \times g$ provides both problems f and g in parallel. For each instance one obtains solutions of both f and g.
3. The coproduct $f \sqcup g$ provides both problems f and g as alternatives. For each instance one can select to obtain either a solution of f or a solution of g.
4. The meet $f \sqcap g$ provides either f or g. For each instance one either obtains a solution for f or a solution for g; one learns a posteriori which one it is, but one cannot control in advance which one it will be.
5. The sum $f + g$ provides two potential solutions for given instances of f and g, at least one of which has to be correct.
6. The finite parallelization f^* allows arbitrarily many finite applications of f in parallel, and with each instance one can select how many applications are to be used in parallel.
7. The parallelization $\widehat{f}$ allows countably many applications of f in parallel.

Given the above list of operations we can derive other algebraic operations.

Definition 11.1.3 (Juxtaposition). For $f :\subseteq X \rightrightarrows Y$ and $g :\subseteq X \rightrightarrows Z$ we denote by $(f,g) :\subseteq X \rightrightarrows Y \times Z$ the *juxtaposition* of f and g, which is defined by $(f,g) := (f \times g) \circ \Delta_X$, where $\Delta_X : X \rightrightarrows X \times X, x \mapsto (x,x)$ denotes the *diagonal* of X.

Given two problems f and g we want to express what it means that f solves g.

Definition 11.1.4 (Solutions). Let $f,g :\subseteq X \rightrightarrows Y$ be multi-valued functions. We define $f \sqsubseteq g :\iff \mathrm{dom}(g) \subseteq \mathrm{dom}(f)$ and $(\forall x \in \mathrm{dom}(g))\ f(x) \subseteq g(x)$. In this situation we say that f *solves* g, f is a *strengthening* of g and g is a *weakening* of f.

Intuitively, $f \sqsubseteq g$ means that all instances of g are also instances of f, and on all these common instances f yields a possible solution of g. It is clear that the relation $\sqsubseteq$ yields a preorder, i.e., it is reflexive and transitive.

Many theorems give rise to mathematical problems. In general, a theorem of the logical form

$$(\forall x \in X)(x \in D \Longrightarrow (\exists y \in Y)P(x,y))$$

translates into the problem

[1] The way we define composition turns the multi-valued functions into morphisms of a specific category [96] that is not identical to the usual category of relations.

$$F :\subseteq X \rightrightarrows Y, x \mapsto \{y \in Y : P(x,y)\} \text{ with } \mathrm{dom}(F) := D.$$

That is, F plays the rôle of a multi-valued Skolem function for the statement of the theorem. The problem F measures the difficulty of finding a suitable y, given x, whereas the condition encapsulated in D is a purely classical premise that is not meant to bear any constructive content. As an example we mention the intermediate value theorem.

Example 11.1.5 (Intermediate value theorem). $\mathsf{IVT} :\subseteq \mathscr{C}[0,1] \rightrightarrows [0,1], f \mapsto f^{-1}\{0\}$, where $\mathrm{dom}(\mathsf{IVT})$ contains all $f \in \mathscr{C}[0,1]$ with $f(0) \cdot f(1) < 0$, is called the *intermediate value theorem*. It is easy to see that $\mathsf{Z}_{[0,1]} \sqsubseteq \mathsf{IVT}$ holds.

Bibliographic Remarks

Algebraic operations on multi-valued functions have been used frequently in computable analysis. For instance, composition in the way defined here, product, juxtaposition and parallelization have been used by Brattka [6, 8]. The coproduct operation and finite parallelization were introduced by Pauly in [91]. The meet operation was introduced by Brattka and Gherardi [18], and the box sum was introduced by Dzhafarov [42]. Inspired by the definition of the box sum, the definition of the sum from Brattka, Gherardi and Hölzl [19] appears here for the first time in a modified version that has better properties. The category of multi-valued functions was studied by Pauly [96].

11.2 Represented Spaces

In this section we want to provide the data types that we will use for problems $f :\subseteq X \rightrightarrows Y$. For a purely topological development of our theory it would be sufficient to consider topological spaces X and Y. However, since we want to discuss computability properties too, we need slightly more structure on the spaces X and Y, and this structure is provided by representations.

Definition 11.2.1 (Represented spaces). A *represented space* (X, δ) is a set X together with a surjective partial function $\delta :\subseteq \mathbb{N}^{\mathbb{N}} \to X$.

If $\delta(p) = x$ then we call p a *name* for x, and we reserve the word *representation* for the map δ itself. We endow the Baire space $\mathbb{N}^{\mathbb{N}}$ with its usual product topology of the discrete topology on $\mathbb{N}$ and we always assume that a represented space (X, δ_X) is endowed with the final topology $\mathscr{O}(X)$ induced by δ_X on X, which is the largest topology on X that turns δ_X into a continuous map. In this situation δ_X is automatically a quotient map. Typically, we will deal with *admissible* representations[2] δ_X that are not just quotient maps but they are even more closely linked to the topology $\mathscr{O}(X)$. In the following we will often just write for short X for a represented space if the representation is clear from the context or not needed explicitly. We can now formally define problems.

[2] See the chapter "*Admissibly Represented Spaces and* Qcb*-Spaces*" by Schröder in this book for more details.

Definition 11.2.2 (Problems). We call partial multi-valued functions $f :\subseteq X \rightrightarrows Y$ on represented spaces X, Y *problems*, for short.

Properties of problems such as computability and continuity can easily be introduced via realizers.

Definition 11.2.3 (Realizer). Given represented spaces (X, δ_X), (Y, δ_Y), a problem $f :\subseteq X \rightrightarrows Y$ and a function $F :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$, we define $F \vdash f :\iff \delta_Y F \sqsubseteq f\delta_X$. In this situation we say that F is a *realizer* of f.

In other words, F is a realizer of f if $\delta_Y F$ solves $f\delta_X$. Obviously, this concept depends on the underlying represented spaces and the notation $F \vdash f$ is only justified when these are clear from the context.

On the Baire space $\mathbb{N}^{\mathbb{N}}$ it is clear what a continuous function $F :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ is. Computability of such functions can be defined via Turing machines in a well-known way. Such properties can now easily be transferred to problems via realizers.

Definition 11.2.4 (Computability and continuity). A problem f is called *computable* (*continuous*) if it has a computable (continuous) realizer.

We warn the reader that the resulting notion of continuity for single-valued functions is not automatically the topological notion of continuity that is induced by the final topologies of the representations. However, every total single-valued function $f : X \to Y$ on represented spaces that is continuous in our sense is also continuous in the usual topological sense with respect to the final topologies, and in all our applications we will use admissible representations for which these two notions even coincide.

Two representations δ_1, δ_2 of the same set X are called *equivalent* if the identity $\mathrm{id} : (X, \delta_1) \to (X, \delta_2)$ and its inverse are computable. It is easy to see that equivalent representations yield the same notion of computability and continuity.

By $\mathscr{C}(X,Y)$ we denote the set of continuous functions $f : X \to Y$ in terms of Definition 11.2.4. The category of represented spaces is Cartesian closed, and the same holds for the category of admissibly represented spaces. In particular, we have canonical ways of defining product and function space representations.

In order to define those, we use pairing functions. We define a pairing function $\langle , \rangle : \mathbb{N}^{\mathbb{N}} \times \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ by $\langle p, q\rangle(2n) := p(n)$ and $\langle p, q\rangle(2n+1) := q(n)$ for $p, q \in \mathbb{N}^{\mathbb{N}}$ and $n \in \mathbb{N}$. We define a pairing function of type $\langle , \rangle : (\mathbb{N}^{\mathbb{N}})^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ by $\langle p_0, p_1, p_2, ...\rangle\langle n, k\rangle := p_n(k)$ for all $p_i \in \mathbb{N}^{\mathbb{N}}$ and $n, k \in \mathbb{N}$, where $\langle n, k\rangle$ is the standard Cantor pairing defined by $\langle n, k\rangle := \frac{1}{2}(n+k+1)(n+k)+k$. Finally, we note that by np we denote the concatenation of a number $n \in \mathbb{N}$ with a sequence $p \in \mathbb{N}^{\mathbb{N}}$.

We assume that we have some standard representation Φ of (a sufficiently large class[3]) of continuous functions, i.e., for any such function $f :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ there is a $p \in \mathbb{N}^{\mathbb{N}}$ with $f = \Phi_p$. For total functions this representation yields the exponential in the category of (admissibly) represented spaces and satisfies natural versions of the

[3] It suffices to consider all continuous functions $f :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ with G_δ–domain since any continuous function can be extended to such a function.

utm- and smn-theorems. For computable p one obtains the computable functions Φ_p with natural domains (see [116] for details). For $p \in \mathbb{N}^{\mathbb{N}}$ we denote by $p-1 \in \mathbb{N}^{\mathbb{N}} \cup \mathbb{N}^*$ the sequence or word that is formed by concatenation of $p(0)-1$, $p(1)-1$, $p(2)-1$,... with the understanding that $-1 = ()$ is the empty word.

Definition 11.2.5 (Constructions on representation). Let (X, δ_X) and (Y, δ_Y) be represented spaces. We define

1. $\delta_{X\times Y} :\subseteq \mathbb{N}^{\mathbb{N}} \to X \times Y$, $\delta_{X\times Y}\langle p,q\rangle := (\delta_X(p), \delta_Y(q))$
2. $\delta_{X\sqcup Y} :\subseteq \mathbb{N}^{\mathbb{N}} \to X \sqcup Y$, $\delta_{X\sqcup Y}(0p) := (0, \delta_X(p))$ and $\delta_{X\sqcup Y}(1p) := (1, \delta_Y(p))$
3. $\delta_{X^*} :\subseteq \mathbb{N}^{\mathbb{N}} \to X^*$, $\delta_{X^*}(n\langle p_1, p_2, ..., p_n\rangle) := (n, (\delta_X(p_1), \delta_X(p_2), ..., \delta_X(p_n)))$
4. $\delta_{X^{\mathbb{N}}} :\subseteq \mathbb{N}^{\mathbb{N}} \to X^{\mathbb{N}}$, $\delta_{X^{\mathbb{N}}}\langle p_0, p_1, p_2, ...\rangle := (\delta_X(p_n))_{n\in\mathbb{N}}$
5. $\delta_{\mathscr{C}(X,Y)} :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathscr{C}(X,Y)$ by $\delta_{\mathscr{C}(X,Y)}(p) = f :\iff \Phi_p \vdash f$
6. $\delta_{\overline{X}} : \mathbb{N}^{\mathbb{N}} \to \overline{X}$, $\delta_{\overline{X}}(p) := \delta_X(p-1)$ if $p-1 \in \mathrm{dom}(\delta_X)$ and $\delta_{\overline{X}}(p) := \bot$ otherwise.

Many spaces that occur in analysis are actually computable metric spaces. For the definition we assume that the reader knows the notion of a computable (double) sequence of real numbers.

Definition 11.2.6 (Computable metric spaces and Cauchy representations).

1. A *computable metric space* (X, d, α) is a separable metric space (X, d) with metric $d : X \times X \to \mathbb{R}$ and a dense sequence $\alpha : \mathbb{N} \to X$ such that $d \circ (\alpha \times \alpha) : \mathbb{N}^2 \to \mathbb{R}$ is a computable double sequence of real numbers.
2. We define the *Cauchy representation* $\delta_X :\subseteq \mathbb{N}^{\mathbb{N}} \to X$ by $\delta_X(p) := \lim_{n\to\infty} \alpha p(n)$ and $\mathrm{dom}(\delta_X) = \{p \in \mathbb{N}^{\mathbb{N}} : (\forall i > j)\, d(\alpha p(i), \alpha p(j)) < 2^{-j}\}$.

A standard numbering of the rational numbers $\mathbb{Q}$ and the Euclidean metric yields the standard Cauchy representation $\delta_{\mathbb{R}}$ of real numbers. Cauchy representations are examples of admissible representations, and for such representations continuity in the usual topological sense and continuity defined via realizers coincides. In particular, $\mathscr{C}(\mathbb{R}) := \mathscr{C}(\mathbb{R}, \mathbb{R})$ is the usual set of continuous functions. In the following we often consider $\mathbb{N}, \mathbb{R}, [0,1], 2^{\mathbb{N}}, \mathbb{N}^{\mathbb{N}}$ and similar spaces as computable metric spaces in the straightforward sense without further mentioning this fact. A *computable Banach space* is just a computable metric space that is additionally a Banach space and such that the linear operations are computable. If the space is additionally a Hilbert space, then it is called a *computable Hilbert space*.

A non-metrizable space that we occasionally need is the Sierpiński space $\mathbb{S} = \{0,1\}$, which is endowed with the topology $\mathscr{O}(\mathbb{S}) = \{\emptyset, \mathbb{S}, \{1\}\}$. By $\widehat{n} \in \mathbb{N}^{\mathbb{N}}$ we denote the constant sequence with value $n \in \mathbb{N}$.

Definition 11.2.7 (Sierpiński space). Let $\delta_{\mathbb{S}} : \mathbb{N}^{\mathbb{N}} \to \mathbb{S}$ be defined by $\delta_{\mathbb{S}}(\widehat{0}) = 0$ and $\delta_{\mathbb{S}}(p) = 1$ for all $p \neq \widehat{0}$.

We close this section with a discussion of computability properties of subsets. The most important notion for us is that of a *co-c.e. closed set*. Given a computable metric space (X, d, α) we denote by $B(x,r) := \{y \in X : d(x,y) < r\}$ the open ball with center $x \in X$ and radius $r \geq 0$. More specifically, we denote by $B_{\langle n,\langle i,k\rangle\rangle} := B(\alpha(n), \frac{i}{k+1})$ a basic open ball.

Definition 11.2.8 (Co-c.e. closed subsets). Let X be a computable metric space. Then $A \subseteq X$ is called *co-c.e. closed* if $X \setminus A = \bigcup_{n\in\mathbb{N}} B_{p(n)}$ for some computable $p \in \mathbb{N}^{\mathbb{N}}$.

For $X = \mathbb{N}^{\mathbb{N}}$ the co-c.e. closed subsets are also known as Π^0_1*–classes*. By $\mathcal{A}(X)$ we denote the set of closed subsets of a topological space X. The definition of co-c.e. closed subsets of computable metric spaces X directly leads to a representation ψ_- of the set $\mathcal{A}(X)$ defined by $\psi_-(p) := X \setminus \bigcup_{n=0}^{\infty} B_{p(n)}$. We denote the represented space $(\mathcal{A}(X), \psi_-)$ for short by $\mathcal{A}_-(X)$. We now formulate two equivalent characterizations of co-c.e. closed sets. For every set $A \subseteq X$ we denote its *characteristic function* by $\chi_A : X \to \mathbb{S}$, which is defined by $\chi_A(x) = 1 :\iff x \in A$.

Proposition 11.2.9 (Co-c.e. closed sets). *Let X be a computable metric space and let $A \subseteq X$. Then the following are equivalent:*

1. *A is co-c.e. closed,*
2. *$A = f^{-1}\{0\}$ for some computable $f : X \to \mathbb{R}$,*
3. *$\chi_{X\setminus A} : X \to \mathbb{S}$ is computable.*

These equivalences are uniform, i.e., the maps $\mathcal{A}_-(X) \to \mathcal{C}(X,\mathbb{S}), A \mapsto \chi_{X\setminus A}$ and $\mathcal{C}(X) \to \mathcal{A}_-(X), f \mapsto f^{-1}\{0\}$ are computable and admit (in the second case multi-valued) computable right inverses.

The third characterization has the advantage that it is the most general of these three, and it works even for arbitrary represented spaces X. Hence, for such spaces we define ψ_- by $\psi_-(p) = A :\iff \delta_{\mathcal{C}(X,\mathbb{S})}(p) = \chi_{X\setminus A}$. We denote the corresponding represented space $(\mathcal{A}(X), \psi_-)$ also by $\mathcal{A}_-(X)$. Due to Proposition 11.2.9 this notation is consistent with the special definition for computable metric spaces X above. Besides the notion of a co-c.e. closed subset we also need the notion of a co-c.e. compact subset.

Definition 11.2.10 (Computable compact subsets). Let X be a computable metric space and let $K \subseteq X$ be compact.

1. K is called *co-c.e. compact* if $\{\langle\langle n_1,...,n_k\rangle, k\rangle \in \mathbb{N} : K \subseteq \bigcup_{i=1}^{k} B_{n_i}\}$ is c.e.
2. K is called *computably compact* if K is co-c.e. compact and there exists a computable sequence that is dense in K.

Obviously, a computable metric space is computably compact if and only if it is co-c.e. compact. Similarly as in the case of closed sets we can derive a representation κ_- of the set $\mathcal{K}(X)$ of compact subsets that is based on (1) and a representation κ of compact sets that is based on (2). By $\mathcal{K}_-(X)$ we denote the represented space $(\mathcal{K}(X), \kappa_-)$. Once again there is a more general representation that works for arbitrary represented spaces, but we will not formalize this representation here.

Bibliographic Remarks

The theory of representations and of computable functions on represented spaces was developed by Kreitz and Weihrauch [72, 73, 117, 113], who also introduced the notion of an admissible representation. Admissible representations in a more general sense have been further studied by Schröder [103], who also recognized the relevance of the Sierpiński space in this context. Computable metric spaces were first introduced by Lacombe [78]. Represented spaces are used as a basic framework for computable analysis [116, 24, 95]. Computability properties of subsets of computable metric spaces were studied since Lacombe [76, 77], and discussions of corresponding representations can be found in [73, 117, 32, 30, 95]. Proposition 11.2.9 for metric spaces is taken from Brattka and Presser [30, Theorem 3.10, Corollary 3.14]. The mentioned representations for compact subsets of metric spaces are also studied in [30]. Representations of subsets for general represented spaces have been studied by Pauly [95].

11.3 The Weihrauch Lattice

We now want to define Weihrauch reducibility as a way to compare problems with each other. The goal is that $f \leq_{\mathrm{W}} g$ expresses the fact that f can be computed by a single application of g. We will need two variants of such a reducibility. By $\mathrm{id} : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ we denote the *identity* of the Baire space. For other sets X we usually add an index X and write the *identity* as $\mathrm{id}_X : X \to X$. For $F, G :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ we define $\langle F, G\rangle(p) := \langle F(p), G(p)\rangle$.

Definition 11.3.1 (Weihrauch reducibility). Let f and g be problems. We define:

1. $f \leq_{\mathrm{W}} g :\iff (\exists \text{ computable } H, K :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}})(\forall G \vdash g)\, H\langle \mathrm{id}, GK\rangle \vdash f$.
2. $f \leq_{\mathrm{sW}} g :\iff (\exists \text{ computable } H, K :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}})(\forall G \vdash g)\, HGK \vdash f$.

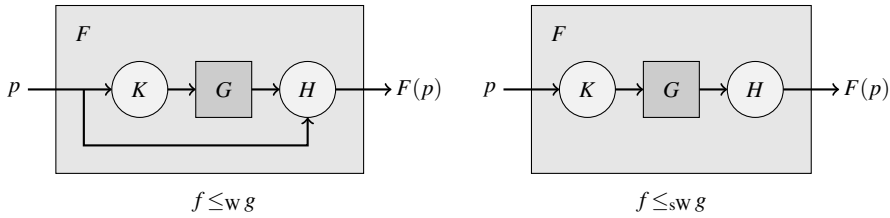
We say that f is *(strongly) Weihrauch reducible* to g, if $f \leq_{\mathrm{W}} g$ ($f \leq_{\mathrm{sW}} g$) holds.

The diagram in Figure 11.1 illustrates Weihrauch reducibility and its strong counterpart. It is easy to see that $f \leq_{\mathrm{sW}} g$ implies $f \leq_{\mathrm{W}} g$. It is also easy to see that $\leq_{\mathrm{W}}$ and $\leq_{\mathrm{sW}}$ are preorders, i.e., they are reflexive and transitive. We denote the corresponding equivalences by $\equiv_{\mathrm{W}}$ and $\equiv_{\mathrm{sW}}$, respectively, and we use the symbols $<_{\mathrm{W}}$ and $<_{\mathrm{sW}}$ for strict reducibilities, respectively. Similar reducibilities can be defined if the notion of computability is replaced by continuity or other suitable categories. A more categorical characterization of Weihrauch reducibility that mentions neither realizers nor the Baire space is given by the following proposition.

Proposition 11.3.2. *Let $f :\subseteq X \rightrightarrows Y$ and $g :\subseteq Z \rightrightarrows W$ be problems. Then:*

1. *$f \leq_{\mathrm{W}} g$ if and only if there are computable $h :\subseteq V \times W \rightrightarrows Y$ and $k :\subseteq X \rightrightarrows V \times Z$ for some represented space V such that $h \circ (\mathrm{id}_V \times g) \circ k \sqsubseteq f$.*
2. *$f \leq_{\mathrm{sW}} g$ if and only if there are computable $h :\subseteq W \rightrightarrows Y$ and $k :\subseteq X \rightrightarrows Z$ such that $h \circ g \circ k \sqsubseteq f$.*

Even though the proof of Proposition 11.3.2 is elementary, there is a subtle point in it. Namely, the proof requires a version of the axiom of choice. In fact, we are freely using the axiom of choice, and mostly we invoke the following version.



$f \leq_{\mathrm{W}} g$ $f \leq_{\mathrm{sW}} g$

Fig. 11.1 Weihrauch reducibility and strong Weihrauch reducibility.

The *axiom of choice for the Baire space*: every problem f has a realizer F.

The fact that Weihrauch reducibility captures the idea of using g exactly once in the course of the computation is stated in the following theorem that we only formulate in intuitive terms here.

Theorem 11.3.3 (Generalized Turing oracles). *$f \leq_{\mathrm{W}} g$ holds if and only if f can be computed on a (generalized) Turing machine that uses exactly one application of g in the course of its computation.*

We emphasize that $f \leq_{\mathrm{W}} g$ actually *requires* that the oracle g is used once in the course of the computation of f. Hence, using the oracle g can actually be an obstacle if the domain of g contains only complicated points.

We note that a characterization of strong Weihrauch reducibility analogous to Theorem 11.3.3 would require discarding all results that were obtained in the course of the computation other than the result of the application of the oracle g. This would be a rather unnatural way of using oracles, and it indicates why ordinary Weihrauch reducibility is a more appropriate concept from this perspective.

The relation between strong and ordinary Weihrauch reducibility is similar to the relation between one-one and many-one reducibility in classical computability theory, and it can be expressed using the notion of a cylinder.

Definition 11.3.4 (Cylinder). A problem f is called a *cylinder* if $\mathrm{id} \times f \leq_{\mathrm{sW}} f$.

It is clear that $f \leq_{\mathrm{sW}} \mathrm{id} \times f$ and $\mathrm{id} \times f \equiv_{\mathrm{W}} f$ hold for all problems f, whereas $\mathrm{id} \times f \leq_{\mathrm{sW}} f$ is a specific property of f that allows one to "feed the input through to f."

Proposition 11.3.5 (Cylinder). *A problem f is a cylinder if and only if for all problems g the following holds: $g \leq_{\mathrm{W}} f \iff g \leq_{\mathrm{sW}} f$.*

It is important to mention that the definitions of $\leq_{\mathrm{W}}$ and $\leq_{\mathrm{sW}}$ are invariant under the replacement of represented spaces by equivalent ones [18, Lemma 2.1]. The equivalence classes induced by $\equiv_{\mathrm{W}}$ and $\equiv_{\mathrm{sW}}$ are called *Weihrauch degrees* and *strong Weihrauch degrees*, respectively. The reducibilities $\leq_{\mathrm{W}}$ and $\leq_{\mathrm{sW}}$ naturally extend to these degrees.

Most algebraic operations defined in Definition 11.1.2 are monotone with respect to (strong) Weihrauch reducibility. We say that a binary operation $\square$ on problems is *monotone* with respect to $\leq_{\mathrm{W}}$ if for all problems f_0, f_1, g_0 and g_1 condition 1. holds, and a unary operation $^{\square}$ on problems is called a *closure operator* with respect to $\leq_{\mathrm{W}}$ if for all problems f, g condition 2. holds:

1. $(f_0 \leq_{\mathrm{W}} f_1$ and $g_0 \leq_{\mathrm{W}} g_1) \Longrightarrow f_0 \square g_0 \leq_{\mathrm{W}} f_1 \square g_1$ (monotone)
2. $f \leq_{\mathrm{W}} f^{\square}, f^{\square\square} \leq_{\mathrm{W}} f^{\square}$ and $(f \leq_{\mathrm{W}} g \Longrightarrow f^{\square} \leq_{\mathrm{W}} g^{\square})$ (closure operator)

Analogously to monotone, we define *antitone* with a reversed order on one side. Monotonicity and closure operators with respect to $\leq_{\mathrm{sW}}$ are defined analogously.

Proposition 11.3.6 (Monotonicity and closure operators). *We obtain:*

1. *The binary operations $\times$, $\sqcup$, $\sqcap$, $\boxplus$ and $+$ are all monotone with respect to $\leq_{\mathrm{W}}$ and $\leq_{\mathrm{sW}}$.*
2. *The unary operation * is a closure operator with respect to $\leq_{\mathrm{W}}$ and monotone with respect to $\leq_{\mathrm{sW}}$.*
3. *The unary operation $\widehat{\ }$ is a closure operator with respect to $\leq_{\mathrm{W}}$ and $\leq_{\mathrm{sW}}$.*

In particular, all the mentioned operations extend to operations on degrees.

It is an obvious question whether there is any least and any greatest Weihrauch degree. The first question is easy to answer.

Definition 11.3.7 (Special Weihrauch degrees). By $\mathbf{0}$ we denote the (strong) Weihrauch degree of the nowhere-defined problems, and by $\mathbf{1}$ we denote the Weihrauch degree of the identity id.

It is easy to see that $\mathbf{0}$ is exactly the class of all nowhere defined problems, and it is the least (strong) Weihrauch degree. The class $\mathbf{1}$ characterizes the computable problems in the sense that $f \leq_{\mathrm{W}} \mathbf{1}$ holds if and only if f is computable. In many respects $\mathbf{0}$ and $\mathbf{1}$ behave algebraically like the numerical constants 0 and 1.

The question of whether there is a greatest Weihrauch degree is less straightforward to answer. If we do not accept the axiom of choice for the Baire space, then the class of problems without realizer forms a natural top element. Since we are accepting the axiom of choice, this natural top element is not available, and we can only add an additional top element to the Weihrauch degrees.[4]

If one is not interested in classifying specific problems with general types X, Y as they appear in analysis, but if one rather wants to study the structure of Weihrauch degrees as such, then it is sufficient to consider problems of the type $f :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}$ on the Baire space. We make this slightly more precise.

Lemma 11.3.8 (Realizer version). *Let (X, δ_X) and (Y, δ_Y) be represented spaces and let $f :\subseteq X \rightrightarrows Y$ be a problem. Then the* realizer version $f^{\mathrm{r}} :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}$ *of f is defined by $f^{\mathrm{r}} := \delta_Y^{-1} \circ f \circ \delta_X$. We have $f^{\mathrm{r}} \equiv_{\mathrm{sW}} f$.*

[4] See Brattka and Pauly [29] for a more detailed discussion.

This means that every (strong) Weihrauch degree has a representative of type $f^{\mathrm{r}} :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}$. By $\mathscr{W}$ and $\mathscr{W}_{\mathrm{s}}$ we denote the set of Weihrauch degrees and strong Weihrauch degrees, respectively, both restricted to problems on the Baire space.[5]

Theorem 11.3.9 (Weihrauch lattice). *The Weihrauch degrees $(\mathscr{W}, \leq_{\mathrm{W}})$ form a distributive lattice with supremum operation $\sqcup$, infimum operation $\sqcap$ and bottom element* $\mathbf{0}$.

Also the strong Weihrauch degrees form a lattice structure, albeit a non-distributive one with a different supremum operation.

Theorem 11.3.10 (Strong Weihrauch lattice). *The strong Weihrauch degrees $(\mathscr{W}_{\mathrm{s}}, \leq_{\mathrm{sW}})$ form a non-distributive lattice with supremum operation $\boxplus$, infimum operation $\sqcap$ and bottom element* $\mathbf{0}$.

Bibliographic Remarks

The reducibility that is now called *Weihrauch reducibility* was introduced by Klaus Weihrauch in the late 1980s and appeared in two unpublished technical reports [114, 115]. He supervised six Ph.D. and M.Sc. projects on this topic (by von Stein [110], Mylatz [83, 84], Brattka [5], Hertling [50], Pauly [90]). Most of this material remained unpublished, and the concept featured only in a few early publications by Brattka, Weihrauch and Gherardi [7, 116, 9, 45, 16]. In a more abstract categorical setting a concept related to Weihrauch reducibility has independently been studied by Hirsch [56]. Theorem 11.3.3 is due to Tavana and Weihrauch [111], and we refer the readers to the reference for a precise formulation. The subject took a turn when Gherardi and Marcone [46] defined Weihrauch reducibility in its full generality for problems, and they promoted Weihrauch complexity as a uniform version of reverse mathematics. Brattka and Gherardi [18, 17] continued to study the subject and discovered that the structure is a lower semilattice, whereas Pauly [92] independently provided the coproduct operation and discovered that the structure is a distributive upper semilattice. The notion of a cylinder is taken from Brattka and Gherardi [18]. Several authors continued to investigate the subject from the perspective of reverse mathematics, among them Dorais, Dzhafarov, Hirst, Mileti and Shafer [38, 40, 42] and Hirschfeldt and Jockusch [57, 58]. Hirschfeldt and Jockusch also introduced a generalized version of Weihrauch reducibility that has a built-in closure under composition. Dzhafarov provided the box sum operation $\boxplus$ and proved that the strong Weihrauch degrees form a non-distributive lattice [42]. Independently, a polynomial-time version of Weihrauch complexity was used by Kawamura and Cook to classify the uniform computational complexity of problems in analysis [65, 66].

11.4 Algebraic and Topological Properties

In this section we discuss a number of algebraic and topological notions and their interactions that turned out to be fruitful for the study of the Weihrauch lattice. We mention that while $f \leq_{\mathrm{W}} \mathbf{1}$ characterizes the computable problems f, also the relation $\mathbf{1} \leq_{\mathrm{W}} f$ bears some meaning.

[5] We use the restriction to the Baire space for our formal definition of $\mathscr{W}$ and $\mathscr{W}_{\mathrm{s}}$, since the class of all (strong) Weihrauch degrees of problems with arbitrary type does not form a set.

Definition 11.4.1 (Pointedness). We call a problem f *pointed* if $\mathrm{id} \leq_{\mathrm{W}} f$ holds. Analogously, we can define *strong pointedness* with the help of $\leq_{\mathrm{sW}}$ instead of $\leq_{\mathrm{W}}$.

It is easy to see that the pointed problems are exactly those with a computable point in their domain. By definition f^* is always pointed since $f^0 = \mathrm{id}_{\{()\}}$. We introduce some further terminology that can be expressed with the help of the algebraic operations.

Definition 11.4.2 (Idempotency and parallelizability). Let f be a problem.

1. We call f *idempotent* if $f \times f \equiv_{\mathrm{W}} f$.
2. We call f *parallelizable* if $\widehat{f} \equiv_{\mathrm{W}} f$.

Analogously, we define *strong idempotency* and *strong parallelizability* with the help of $\equiv_{\mathrm{sW}}$ instead of $\equiv_{\mathrm{W}}$.

Whether or not a problem is idempotent or parallelizable might be hard to prove in some instances. In the following example the first statement is relatively easy to obtain, whereas the second one is harder to prove (see Theorems 11.7.34 and 11.7.35).

Example 11.4.3. $\widehat{\mathsf{IVT}} \equiv_{\mathrm{sW}} \mathsf{Z}_{[0,1]}$ and hence $\mathsf{Z}_{[0,1]}$ is (strongly) parallelizable, but IVT is not idempotent.

The following result captures some easy observations. Pointedness is involved here, since $f^0 = \mathrm{id}_{\{()\}}$ is pointed for every problem f.

Proposition 11.4.4 (Idempotency and parallelizability). *Let f be a problem. Then:*

1. *f (strongly) parallelizable $\Longrightarrow$ f (strongly) idempotent.*
2. *f pointed and idempotent $\iff$ $f^* \equiv_{\mathrm{W}} f$.*
3. *f strongly pointed and strongly idempotent $\iff$ $f^* \equiv_{\mathrm{sW}} f$.*

A less obvious result relates idempotency and parallelizability. In order to formulate this result, we need another definition.

Definition 11.4.5 (Finite tolerance). A *problem* $f :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}$ is called *finitely tolerant* if there is a computable partial function $T :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ such that for all $p,q \in \mathrm{dom}(f)$ and $k \in \mathbb{N}$ with $(\forall n \geq k)(p(n) = q(n))$ it follows that $r \in f(q)$ implies $T\langle r,k\rangle \in f(p)$. More generally, a problem $g :\subseteq X \rightrightarrows Y$ can be called *finitely tolerant* if there is some finitely tolerant $f :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}$ with $f \equiv_{\mathrm{W}} g$.

Intuitively, finite tolerance means that for two almost identical inputs and a solution for one of these inputs we can compute a solution for the other input. The squashing theorem relates products $g \times f$ to parallelizations $\widehat{g}$ of problems.

Theorem 11.4.6 (Squashing theorem). *For $f,g :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}$ we obtain:*

1. *If $\mathrm{dom}(f) = \mathbb{N}^{\mathbb{N}}$ and f is finitely tolerant, then $g \times f \leq_{\mathrm{W}} f \Longrightarrow \widehat{g} \leq_{\mathrm{W}} f$.*

2. *If* $\mathrm{dom}(f) = 2^{\mathbb{N}}$ *and* f *is finitely tolerant, then* $g \times f \leq_{\mathrm{sW}} f \Longrightarrow \widehat{g} \leq_{\mathrm{sW}} f$.

We obtain the following immediate corollary.

Corollary 11.4.7. *Let* $f :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}$ *be finitely tolerant. Then we obtain:*

1. *For* $\mathrm{dom}(f) = \mathbb{N}^{\mathbb{N}}$*:* f *idempotent* $\iff$ f *parallelizable.*
2. *For* $\mathrm{dom}(f) = 2^{\mathbb{N}}$*:* f *strongly idempotent* $\iff$ f *strongly parallelizable.*

Another property that turned out to be quite useful is join-irreducibility. We recall that a problem f is called *join-irreducible* in the lattice-theoretic sense if $f \leq_{\mathrm{W}} g \sqcup h$ implies $f \leq_{\mathrm{W}} g$ or $f \leq_{\mathrm{W}} h$ for all problems g, h. We need a countable version of this property. For this purpose we first need to define countable coproducts. For a sequence $(X_i)_{i \in \mathbb{N}}$ of sets we define the disjoint union by $\bigsqcup_{i=0}^{\infty} X_i := \bigcup_{i=0}^{\infty} (\{i\} \times X_i)$. Now we can define the countable coproduct.

Definition 11.4.8 (Countable coproduct). Let $f_i :\subseteq X_i \rightrightarrows Y_i$ be problems for all $i \in \mathbb{N}$. Then we define $\bigsqcup_{i=0}^{\infty} f_i :\subseteq \bigsqcup_{i=0}^{\infty} X_i \rightrightarrows \bigsqcup_{i=0}^{\infty} Y_i$ by $\bigsqcup_{i=0}^{\infty} f_i(n,x) := \{n\} \times f_n(x)$.

Now we are prepared to define countable irreducibility.

Definition 11.4.9 (Countable irreducibility). A problem f is called *countably irreducible* if for every sequence $(g_i)_{i \in \mathbb{N}}$ of problems: $f \leq_{\mathrm{W}} \bigsqcup_{i=0}^{\infty} g_i \Longrightarrow (\exists i) f \leq_{\mathrm{W}} g_i$. Likewise we can define *strong countable irreducibility* with $\leq_{\mathrm{sW}}$ in place of $\leq_{\mathrm{W}}$.

It is clear that every countably irreducible[6] problem is join-irreducible. Another notion that turned out to be fruitful in this context is the notion of a fractal. Roughly speaking, a fractal is a problem that exhibits its full power even if we zoom arbitrarily deep into its domain.

Definition 11.4.10 (Fractal). A problem f is called a *fractal* if there is a problem $F :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}$ such that $F \equiv_{\mathrm{W}} f$ and $F|_A \equiv_{\mathrm{W}} F$ holds for every clopen $A \subseteq \mathbb{N}^{\mathbb{N}}$ with $A \cap \mathrm{dom}(F) \neq \emptyset$. Likewise we define a *strong fractal* with $\equiv_{\mathrm{sW}}$ instead of $\equiv_{\mathrm{W}}$. A *total (strong) fractal* is a (strong) fractal where F can be chosen to be total.

One reason why fractals are useful is captured in the following observation.

Proposition 11.4.11 (Fractals). *Every (strong) fractal is (strongly) countably irreducible.*

Some natural problems in the Weihrauch lattice are densely realized in the following sense.

Definition 11.4.12 (Densely realized). Let (X, δ_X), (Y, δ_Y) be represented spaces. A problem $f :\subseteq X \rightrightarrows Y$ is called *densely realized* if $f^{\mathrm{r}}(p) = \delta_Y^{-1} \circ f \circ \delta_X(p)$ is dense in $\mathrm{dom}(\delta_Y)$ for all $p \in \mathrm{dom}(f \circ \delta_X)$.

[6] We note that countable irreducibility is not identical to what is sometimes called σ*-join-irreducibility* since the countable coproduct is not necessarily a countable supremum, as we will see in Theorem 11.5.1.

We note that this notion depends on the representations chosen. It turns out that all problems with discrete output below densely realized problems with totally represented output are computable.

Proposition 11.4.13 (Densely realized). *Let $f :\subseteq X \rightrightarrows Y$ be densely realized, where Y is a represented space with total representation, and let $g :\subseteq Z \rightrightarrows \mathbb{N}$ be a problem. If $g \leq_{\mathrm{W}} f$ holds, then g is computable.*

Bibliographic Remarks

The notions of pointedness, idempotency and parallelizability were introduced by Brattka and Gherardi in [18]. Dorais, Dzhafarov, Hirst, Mileti and Shafer [38] defined finitely tolerant problems and proved the squashing theorem (a proof for Theorem 11.4.6 exactly as stated here can be found in [102]). Countable irreducibility was first considered by Brattka, de Brecht and Pauly [14], who also implicitly defined fractals that were later used by Brattka, Gherardi and Marcone [21]. Densely realized problems have been introduced by Brattka, Hendtlass and Kreuzer [22] and Proposition 11.4.13 is due to Brattka and Pauly [29].

11.5 Completeness, Composition and Implication

Another obvious question regarding the Weihrauch lattice is whether the lattice is complete or, more generally, which suprema and infima exist. A mostly negative answer is given by the following result.

Theorem 11.5.1 (Suprema and infima). *No non-trivial countable suprema exist in the Weihrauch lattice, i.e., a sequence $(f_n)_{n\in\mathbb{N}}$ of problems has a supremum if and only if this supremum is already a supremum of $(f_n)_{n\leq k}$ for some $k \in \mathbb{N}$. Some non-trivial countable infima exist in the Weihrauch lattice, others do not exist.*

In particular, the Weihrauch lattice is not complete in the lattice-theoretic sense. We can also conclude that $\bigsqcup_{n=0}^{\infty} f_n$ is typically not the supremum of $\{f_n : n \in \mathbb{N}\}$ unless it is already a supremum of $\bigsqcup_{n=0}^{k} f_n$ for some $k \in \mathbb{N}$.

However, it turns out that some important suprema and infima exist in the Weihrauch lattice. We are particularly interested in composition and implication. The composition $f \circ g$ of problems as it has been defined in Definition 11.1.2 is not an operation on degrees in the same sense in which the other algebraic operations extend to degrees. It requires that the output type of g fits the input type of f, and even if the types fit, the operation does not need to be monotone. On the other hand, it is natural to consider a Weihrauch degree $f * g$ that captures exactly what can be achieved when one first applies g, possibly followed by some computation, and then one applies f. That the maximal Weihrauch degree that can be built in this way always exists is the first statement of the following theorem. The second statement captures the minimal degree $(g \to f)$ that is needed in advance of g in order to compute f. In some sense $(g \to f)$ measures how much harder f is to compute than g.

Theorem 11.5.2 (Compositional product and implication). *Let f and g be problems. The following Weihrauch degrees exist:*

1. $f * g := \max_{\leq_W}\{f_0 \circ g_0 : f_0 \leq_W f, g_0 \leq_W g\}$ *(compositional product)*
2. $(g \to f) := \min_{\leq_W}\{h : f \leq_W g * h\}$ *(implication)*

Maximum and minimum are understood with respect to $\leq_W$. Only such f_0 and g_0 are considered that can be composed.

By definition, $*$ and $\to$ are operations on degrees. It is easy to see that $*$ is even a monotone operation, whereas $\to$ is antitone in the first component and monotone in the second component. In order to prove Theorem 11.5.2 it is useful to define a specific representative of the degree $f * g$ that we denote by $f \star g$. For $F, G :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ we define $\langle F \times G\rangle\langle p, q\rangle := \langle F(p), G(q)\rangle$.

Definition 11.5.3 (Compositional product). Let f and g be problems. We define $f \star g :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}$ by $(f \star g)\langle p, q\rangle := \langle \mathrm{id} \times f^{\mathrm{r}}\rangle \circ \Phi_p \circ g^{\mathrm{r}}(q)$ for all $p, q \in \mathbb{N}^{\mathbb{N}}$.

This definition captures the intuition that in between g and f there is another possible computation Φ_p. Of course, this definition does not have the same set-theoretic flavor as the other operations in Definition 11.1.2, and it is not a definition that we typically work with. It is mostly needed in order to prove Theorem 11.5.2, and the working definition of the compositional product $f * g$ is the one given in Theorem 11.5.2. The following result captures another interesting property of $f \star g$.

Proposition 11.5.4. *$f * g \equiv_W f \star g$ and $f \star g$ is always a cylinder. If f and g are fractals, then so is $f \star g$.*

We can also define a strong version of the compositional product. This operation has been studied less and is only known to exist in specific cases. In fact, since $f \star g$ is always a cylinder, we directly obtain the following corollary of Theorem 11.5.2 and Proposition 11.5.4.

Corollary 11.5.5. *$f *_{\mathrm{s}} g := \max_{\leq_{\mathrm{sW}}}\{f_0 \circ g_0 : f_0 \leq_{\mathrm{sW}} f, g_0 \leq_{\mathrm{sW}} g\}$ exists for cylinders f, g.*

The maximum $f *_{\mathrm{s}} g$ exists also in some cases where f, g are not cylinders, but we do not claim that it exists in general. The following result summarizes some algebraic properties of compositional products and implications.

Proposition 11.5.6 (Algebraic properties).

1. *$*$ is associative but not commutative, $\to$ is neither associative nor commutative.*
2. *$*_{\mathrm{s}}$ is associative whenever all occurring degrees actually exist.*

The operations $+, \sqcap, \boxplus, \sqcup, \times$ and $*$ are typically ordered as given.

Proposition 11.5.7 (Order of algebraic operations). *We obtain:*

1. *$f + g \leq_{\mathrm{sW}} f \sqcap g \leq_{\mathrm{sW}} f \boxplus g \leq_{\mathrm{sW}} f \sqcup g$ and $f \times g \leq_{\mathrm{sW}} f \star g$ for all problems f, g.*

2. $f \sqcup g \leq_{\mathrm{W}} f \times g$ *and* $f \sqcap\!\sqcap g \leq_{\mathrm{sW}} f \times g$ *for all pointed problems* f, g.

The following result expresses in which way compositional product and implication are adjoints of each other.

Proposition 11.5.8 (Adjointness). $f \leq_{\mathrm{W}} g * h \iff (g \to f) \leq_{\mathrm{W}} h$.

In the language of lattice theory this result can be expressed such that $(\mathscr{W}, \geq_{\mathrm{W}}, *)$ is right residuated, and the residual operation is exactly $\to$. It follows from Example 11.7.43 that $(\mathscr{W}, \geq_{\mathrm{W}}, *)$ is not left residuated and that $(\mathscr{W}, \geq_{\mathrm{W}}, \times)$ is not residuated. The following result expresses that the Weihrauch lattice is not residuated with respect to the lattice operations $\sqcup, \sqcap$.

Theorem 11.5.9 (Brouwer and Heyting algebras). *The Weihrauch lattice* $\mathscr{W}$ *is neither a Brouwer algebra nor a Heyting algebra.*

Brouwer algebras can be seen as models of intermediate logics that are in between classical logic and intuitionistic logic.

Bibliographic Remarks

Theorems 11.5.1 and 11.5.9 are due to Higuchi and Pauly [55], who also discussed several variants of the Weihrauch lattice in this regard. The compositional product was introduced by Brattka, Gherardi and Marcone [21]. The implication was introduced by Brattka and Pauly [95], who also proved most other results in this section. They also studied many further algebraic properties of the Weihrauch lattice, including distributivity laws. The results on strong compositional products are taken from Brattka, Hendtlass and Kreuzer [22].

11.6 Limits and Jumps

A map of particular importance in the Weihrauch lattice is the limit map. Given a Hausdorff space X, we define the *limit map of the space* X and the *the limit map* (of the Baire space) by

1. $\lim_X :\subseteq X^{\mathbb{N}} \to X, (x_n)_{n\in\mathbb{N}} \mapsto \lim_{n\to\infty} x_n$,
2. $\lim :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}, \langle p_0, p_1, p_2, ...\rangle \mapsto \lim_{n\to\infty} p_n$.

The domain of $\lim_X$ consists of all converging sequences in X. In the special case of the Baire space, we use a tupling with $p_i \in \mathbb{N}^{\mathbb{N}}$ on the input side for mere reasons of convenience. By $\lim_\Delta$ we denote the restriction of lim to eventually constant sequences. It is easy to see that $\lim_{\mathbb{N}} \equiv_{\mathrm{W}} \lim_\Delta$. It has been noticed that limit maps can be used to characterize limit computable functions and functions computable with finitely many mind changes.

Proposition 11.6.1 (Limit computability and finite mind change computability). *For problems* f *we obtain:*

1. $f \leq_{\mathrm{W}} \lim \iff$ *f limit computable.*
2. $f \leq_{\mathrm{W}} \lim_{\mathbb{N}} \iff$ *f computable with finitely many mind changes.*

Limit computability and computability with finitely many mind changes can be defined directly with Turing machines that allow two-way output tapes. In the case of limit computable problems the Turing machine can change the content of each output cell finitely many times before it has to stabilize; in the case of problems that are computable with finitely many mind changes, the entire output has to stabilize after finitely many changes. These concepts are well known from learning theory.

One might ask whether $\lim_X$ for other spaces X yields different classes of computable problems, but for many spaces X this is not the case. We recall that a computable metric space X is called *rich* if there is a *computable embedding* $\iota : 2^{\mathbb{N}} \hookrightarrow X$, i.e., ι is injective, and ι and its partial inverse ι^{-1} are computable.

Proposition 11.6.2 (Limits). $\lim_X \equiv_{\mathrm{sW}} \lim$ *for all rich computable metric spaces X.*

Examples of rich computable metric spaces are $2^{\mathbb{N}}, \mathbb{N}^{\mathbb{N}}, \mathbb{R}, \mathbb{R}^{\mathbb{N}}, [0,1], [0,1]^{\mathbb{N}}$, etc. This justifies also the more generic notation lim for the limit operation on the Baire space. An interesting property of $\lim_X$ is its behavior under composition. The following result on $\lim_{\mathbb{N}}$ can be proved with the help of Theorem 11.7.11, but is also easy to see directly.

Proposition 11.6.3 (Composition). $\lim_{\mathbb{N}} * \lim_{\mathbb{N}} \equiv_{\mathrm{W}} \lim_{\mathbb{N}}$*, i.e., problems that are computable with finitely many mind changes are closed under composition.*

The situation for iterations of lim is very different. For a problem f we denote by $f^{[n]}$ the n–fold iteration of the compositional product of f with itself, i.e., $f^{[0]} \equiv_{\mathrm{W}} \mathrm{id}$, $f^{[1]} \equiv_{\mathrm{W}} f$, $f^{[2]} \equiv_{\mathrm{W}} f * f$, etc. By iterations of lim one climbs up the Borel hierarchy with every further application of a limit. By Σ^0_n we denote the corresponding Borel class of subsets of $\mathbb{N}^{\mathbb{N}}$, i.e., Σ^0_1 is the class of open subsets, Σ^0_2 is the class of F_σ–subsets and so forth. A function $F :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ is called *Σ^0_n–measurable*, if preimages $F^{-1}(U)$ of open sets U are Σ^0_n–sets relative to $\mathrm{dom}(F)$. Analogously, F is *effectively Σ^0_n–measurable* if the preimage can be uniformly computed from a description of U. The Σ^0_1–measurable functions F are exactly the continuous ones, and the effectively Σ^0_1–measurable functions F are exactly the computable ones. We can transfer concepts of measurability to problems via realizers.

Definition 11.6.4 (Effective Borel measurability). Let $n \geq 1$. A problem f is called *(effectively) Σ^0_n–measurable* if it has a realizer with the same property.

It can be proved that for computable metric spaces X and Y and total functions $f : X \to Y$ this yields just the usual (effectively) Σ^0_n–measurable functions as they are known in descriptive set theory [9]. The measurable problems can also easily be characterized in the Weihrauch lattice.

Theorem 11.6.5 (Effective Borel measurability). $f \leq_{\mathrm{W}} \lim^{[n]} \iff$ *f is effectively Σ^0_{n+1}–measurable, for all problems f and $n \in \mathbb{N}$.*

This theorem can be relativized. We write $f \leq_{\mathrm{W}}^{p} g$ if f is Weihrauch reducible to g with respect to some oracle $p \in \mathbb{N}^{\mathbb{N}}$, which means that the reduction functions H, K are computable relative to p. Then $f \leq_{\mathrm{W}}^{p} \lim^{[n]}$ holds for some $p \in \mathbb{N}^{\mathbb{N}}$ if and only if f is Σ^0_{n+1}–measurable. Theorem 11.6.5 shows that the Weihrauch lattice yields a refinement of the effective Borel hierarchy very much in the same way as many-one reducibility yields a refinement of the Kleene hierarchy. We summarize some of the obvious algebraic properties of lim.

Proposition 11.6.6. lim *is a cylinder, strongly parallelizable, strongly idempotent, finitely tolerant, a strong fractal and (strongly) countably irreducible.*

It is useful to know that there are many problems that are equivalent to lim. We mention only a few. By $\mathsf{J} : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}, p \mapsto p'$ we denote the Turing jump operation, which is injective as a function on the Baire space. By $\mathsf{EC} : \mathbb{N}^{\mathbb{N}} \to 2^{\mathbb{N}}$ we denote the function that translates enumerations of sets into their characteristic functions, and by $\mathsf{LPO} : \mathbb{N}^{\mathbb{N}} \to \{0,1\}$ we denote the *limited principle of omniscience*[7]:

$$\mathsf{EC}(p)(n) := \begin{cases} 1 \text{ if } (\exists k \in \mathbb{N})\ p(k) = n+1 \\ 0 \text{ otherwise} \end{cases} \quad \text{and } \mathsf{LPO}(p) := \begin{cases} 1 \text{ if } (\exists k)\ p(k) = 0 \\ 0 \text{ otherwise} \end{cases}$$

We also use $\inf, \sup :\subseteq \mathbb{R}^{\mathbb{N}} \to \mathbb{R}$. We obtain the following result that lists some important members of the equivalence class of lim.

Theorem 11.6.7 (Limit). $\lim \equiv_{\mathrm{sW}} \inf \equiv_{\mathrm{sW}} \sup \equiv_{\mathrm{sW}} \mathsf{J} \equiv_{\mathrm{sW}} \mathsf{EC} \equiv_{\mathrm{sW}} \widehat{\mathsf{LPO}} \equiv_{\mathrm{sW}} \widehat{\lim_{\mathbb{N}}}$.

We now use this limit operation to define the jump of a represented space.

Definition 11.6.8 (Jump of a represented space). Let (X, δ) be a represented space. Then we define its *jump* (X', δ') by $X' := X$ and $\delta' := \delta \circ \lim$. Likewise, $(X^{(n)}, \delta^{(n)})$ denotes the n–fold jump.

That is, in the new represented space X' a name of x with respect to δ' is a sequence that converges to a name in the sense of δ. Hence, names in (X', δ') typically carry less computably accessible information than names in (X, δ). Now the jump of a problem is just the same problem but with the input space replaced by its jump.

Definition 11.6.9 (Jump of a problem). Let $f :\subseteq X \rightrightarrows Y$ be a problem. Then its *jump* $f' :\subseteq X' \rightrightarrows Y$ is defined to be the same problem with the modified input space X'. Likewise $f^{(n)} :\subseteq X^{(n)} \rightrightarrows Y$ denotes the n–fold jump for $n \in \mathbb{N}$ with $f^{(0)} := f$.

Since the jump f' has to work with a weaker type of input information, it is typically harder to compute f' than f. The study of jumps provides one reason why it is important to keep track of strong Weihrauch reductions. Jumps are monotone with respect to strong Weihrauch reductions, but not with respect to ordinary Weihrauch reductions in general.

[7] LPO is also Weihrauch equivalent to the identity $\mathrm{id} : \mathbb{S} \to \{0,1\}$.

Proposition 11.6.10 (Monotonicity). *For all problems f,g the following hold: $f \leq_{\mathrm{sW}} f'$ and also $f \leq_{\mathrm{sW}} g \Longrightarrow f' \leq_{\mathrm{sW}} g'$.*

Trivial examples such as a constant function show that $f \equiv_{\mathrm{sW}} f'$ can happen. For $f \leq_{\mathrm{W}} g$ all possible reductions between f' and g' can occur; the order can even be reversed, i.e., $g' <_{\mathrm{W}} f'$ can happen [22, Figure 2]. Surprisingly, there is also a certain inverse of Proposition 11.6.10 that one can prove. We recall that by p' we denote the Turing jump of p. Using this concept we can phrase the following theorem.

Theorem 11.6.11 (Inverting jumps). *$f' \leq_{\mathrm{W}}^{p} g' \Longrightarrow f \leq_{\mathrm{W}}^{p'} g$ holds for all problems f,g and $p \in \mathbb{N}^{\mathbb{N}}$. An analogous statement holds if Weihrauch reducibility is replaced by strong Weihrauch reducibility in both occurrences.*

The property that $f \leq_{\mathrm{W}}^{p} g$ holds for some oracle p is equivalent to the continuous version of Weihrauch reducibility, where the two reduction functions H,K just need to be continuous. By Theorem 11.6.11 continuous separations are particularly useful, since they automatically carry over to jumps.

The following result summarizes some algebraic properties of the jump.

Proposition 11.6.12 (Algebraic properties). *We obtain $f' \times g' \equiv_{\mathrm{sW}} (f \times g)'$, $\widehat{f'} \equiv_{\mathrm{sW}} \widehat{f}'$, $f' \sqcap g' \equiv_{\mathrm{sW}} (f \sqcap g)'$, $f' \sqcup g' \leq_{\mathrm{sW}} (f \sqcup g)'$ and $f'^{*} \leq_{\mathrm{sW}} f^{*\prime}$ for all problems f,g.*

One can see that coproducts do not commute with jumps in general, since jumps are join-irreducible.

Proposition 11.6.13 (Finite tolerance and fractality). *f' is finitely tolerant, a strong fractal and (strongly) countably irreducible for every problem f.*

Sometimes it is useful to have the following characterizations of the jump.

Proposition 11.6.14 (Cylinder). *$f' \equiv_{\mathrm{sW}} f *_{\mathrm{s}} \lim$, and if f is a cylinder, then f' is a cylinder and $f' \equiv_{\mathrm{W}} f' \times \lim \equiv_{\mathrm{W}} f * \lim$.*

In particular, $f *_{\mathrm{s}} \lim$ always exists. We continue with a discussion of some invariant properties. We call a class P of problems *invariant* if $f \leq_{\mathrm{W}} g$ and $g \in P$ implies $f \in P$. Likewise, we define *strong invariance*. We list a number of examples of (strongly) invariant properties that easily follow from results of this section.

Corollary 11.6.15 (Invariance). *The following properties of problems are (strongly) invariant: continuity, computability, limit computability, (effective) Σ^0_n–measurability, computability with finitely many mind changes, non-uniform computability (i.e., the class of problems that have some computable output for every computable input in the domain).*

Sometimes it is also useful to use numerical quantities that are preserved by Weihrauch reducibility. Besides the level n of (effective) Σ^0_n–measurability, we can also use the number of mind changes that are required to compute a problem. Let $\mathrm{mind}(f)$ denote the minimal number $n \in \mathbb{N}$ that a Turing machine with two-way output needs in order to compute f with at most n mind changes on all inputs (if such a number exists). This property is invariant in the following sense.

Proposition 11.6.16 (Mind changes). $f \leq_{\mathrm{W}} g \Longrightarrow \mathrm{mind}(f) \leq_{\mathrm{W}} \mathrm{mind}(g)$ *for all problems* f, g *for which* $\mathrm{mind}(f), \mathrm{mind}(g)$ *exist.*

Another numerical quantity that is useful as an invariant for strong Weihrauch reducibility is the cardinality of a problem.

Definition 11.6.17 (Cardinality). For every problem $f :\subseteq X \rightrightarrows Y$ we denote by $\#f$ the maximal cardinality (if it exists) of a set $M \subseteq \mathrm{dom}(f)$ such that $\{f(x) : x \in M\}$ contains pairwise disjoint sets.

It is easy to see that the following holds.

Proposition 11.6.18 (Cardinality). $f \leq_{\mathrm{sW}} g \Longrightarrow \#f \leq \#g$ *for all problems* f, g *with existing cardinality.*

Every cylinder f needs to satisfy $\#f \geq |\mathbb{N}^{\mathbb{N}}|$, since $\#\mathrm{id} = |\mathbb{N}^{\mathbb{N}}|$, where $|X|$ denotes the *cardinality* of the set X. For instance, it is easy to see that $\#\lim_{\mathbb{N}} = |\mathbb{N}|$ and $\#\lim_{\Delta} = |\mathbb{N}^{\mathbb{N}}|$. We obtain the following.

Example 11.6.19. $\lim_{\mathbb{N}} \equiv_{\mathrm{W}} \lim_{\Delta}$ and $\lim_{\mathbb{N}} <_{\mathrm{sW}} \lim_{\Delta}$. Moreover, $\lim_{\Delta}$ is a cylinder, whereas $\lim_{\mathbb{N}}$ is not.

Next we mention that LPO is in a certain sense the weakest discontinuous problem among all *single-valued* problems.

Theorem 11.6.20 (Discontinuous single-valued problems). $\mathsf{LPO} \leq^{p}_{\mathrm{W}} f$ *for some oracle* $p \iff f$ *is discontinuous, for* $f : X \to Y$ *on computable metric spaces* X, Y.

We close this section with the following result that shows that for (certain well-behaved) linear closed operators there is a dichotomy: either they are bounded and computable or lim is reducible to them.

Theorem 11.6.21 (Linear operators). *Let* $T :\subseteq X \to Y$ *be a linear closed operator on computable Banach spaces* X, Y. *Let* $(e_n)_{n\in\mathbb{N}}$ *be a computable sequence in* $\mathrm{dom}(T)$ *whose linear span is dense in* $\mathrm{dom}(T)$ *and such that* $(T(e_n))_{n\in\mathbb{N}}$ *is computable.*

1. *If* T *is bounded, then* T *is computable.*
2. *If* T *is unbounded, then* $\lim \leq_{\mathrm{W}} T$.

Bibliographic Remarks

The relation between Weihrauch reducibility and the Borel hierarchy was established by Brattka, including Proposition 11.6.2 and Theorem 11.6.5 [9, 5] (see also [31]). Limit computable functions and functions that are computable by finite mind changes were originally introduced to computable analysis by Ziegler [119, 118]. The relation of these classes to the Weihrauch lattice was studied by Brattka and Gherardi [18, 17], including Corollary 11.6.15 and Proposition 11.6.16 and by Brattka, de Brecht and Pauly [14], including Proposition 11.6.3. Theorem 11.6.7 collects results from the last-mentioned groups of authors. The concept of a jump of a problem was introduced by Brattka,

Gherardi and Marcone [21], who also proved most related results that are included here, except Theorem 11.6.11, which is due to Brattka, Hölzl and Kuyper [25], and the proof is essentially based on a jump control theorem by Brattka, Hendtlass and Kreuzer [22]. The number of mind changes has a topological counterpart, namely the level of a problem as studied by Hertling [50, 49]; it essentially measures the level in the Hausdorff difference hierarchy. Proposition 11.6.18 is due to Brattka, Gherardi and Hölzl [19, Proposition 3.6]. Theorem 11.6.20 is due to Weihrauch [115, Theorem 3.7]. Theorem 11.6.21 is due to Brattka [7, Theorem 4.3] and can be seen as a uniform version of the first main theorem of Pour-El and Richards [101].

11.7 Choice

The choice problem C_X of a given space X is the problem of finding a point in a given closed $A \subseteq X$. By choosing appropriate spaces X one obtains several important Weihrauch degrees.

Definition 11.7.1 (Choice). The problem $\mathsf{C}_X :\subseteq \mathscr{A}_-(X) \rightrightarrows X, A \mapsto A$ with $\mathrm{dom}(\mathsf{C}_X) := \{A : A \neq \emptyset\}$ is called the *choice problem* of the represented space X.

Here the description $A \mapsto A$ of the map is to be read such that on the input side $A \in \mathscr{A}_-(X)$ is a point of the input space, whereas on the output side it is a subset $A \subseteq X$ of possible results. Typically, X will be a computable metric space, and the reader can think of closed sets being represented by enumerations of balls whose union exhausts the complement, as described in the first item of Proposition 11.2.9.

Likewise, one can use the second characterization of Proposition 11.2.9 to define a representation of closed sets via preimages of continuous functions $f : X \to \mathbb{R}$. The uniformity statement in Proposition 11.2.9 yields the conclusion that the choice problem is nothing but the zero problem that we introduced in Example 11.1.1.

Corollary 11.7.2. $\mathsf{C}_X \equiv_{\mathrm{sW}} \mathsf{Z}_X$ *for every computable metric space* X.

Certain relations between spaces transfer to the corresponding choice problems. We mention two such properties in the following result.

Proposition 11.7.3 (Subsets and surjections). *Let* X, Y *be represented spaces.*

1. *If* $A \subseteq X$ *is co-c.e. closed, then* $\mathsf{C}_A \leq_{\mathrm{sW}} \mathsf{C}_X$.
2. *If there is a computable surjection* $s : X \to Y$, *then* $\mathsf{C}_Y \leq_{\mathrm{sW}} \mathsf{C}_X$.

The choice problem has been studied in many variants that are typically restrictions to closed subsets with certain extra properties. We list a number of examples.

Definition 11.7.4 (Variants of choice).

1. UC_X is C_X restricted to singletons (*unique choice*)
2. CC_X is C_X restricted to connected sets (*connected choice*)
3. PWCC_X is C_X restricted to pathwise connected sets (*pathw. connected choice*)
4. XC_X is C_X restricted to convex sets (*convex choice*)

5. PC_X is C_X restricted to sets with positive measure (*positive choice*)
6. AoUC_X is C_X restricted to sets of the form $\{x\}$ or X (*all-or-unique choice*)
7. ACC_X is C_X restricted to sets of the form $X \setminus \{x\}$ or X (*all-or-co-unique choice*)
8. CFC_X is C_X restricted to co-finite sets (*co-finite choice*)

In some of these examples some additional structure is required on X. For instance, for convex choice one would assume that X is a vector space, and for positive choice one would expect that X is endowed with a fixed Borel measure. In the case of $\mathbb{N}^{\mathbb{N}}$ and $\mathbb{R}$ we assume that the product measure of the geometric probability measure on $\mathbb{N}$ and the Lebesgue measure are used, respectively. In the case of UC_X and AoUC_X we assume that X is a T_1–space, and in the case of ACC_X and CFC_X we assume that X is endowed with a discrete topology. The choice problem C_X is a fractal for many spaces X and often a total fractal for compact X.

Proposition 11.7.5 (Fractality).

1. $\mathsf{C}_{\mathbb{N}}, \mathsf{C}_{\mathbb{R}}, \mathsf{PC}_{\mathbb{R}}$ *and* $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$ *are fractals,*
2. $\mathsf{C}_{2^{\mathbb{N}}}, \mathsf{PC}_{2^{\mathbb{N}}}, \mathsf{CC}_{[0,1]}$ *and* $\mathsf{XC}_{[0,1]^{n+1}}$ *are total fractals for all* $n \in \mathbb{N}$.

In particular, all the mentioned problems are countably irreducible and hence join-irreducible.

While it is easy to see that C_X is a cylinder for many spaces X, it follows from the fact that there are only countably many pairwise different sets of positive measure that $\#\mathsf{PC}_{2^{\mathbb{N}}} = \#\mathsf{PC}_{\mathbb{R}} = \#\mathsf{PC}_{\mathbb{N}^{\mathbb{N}}} = \#\mathsf{C}_{\mathbb{N}} = |\mathbb{N}|$, and hence all the mentioned problems are not cylinders. It requires more sophisticated arguments to show that $\mathsf{CC}_{[0,1]}$ is not a cylinder despite the fact that $\#\mathsf{CC}_{[0,1]} = |\mathbb{N}^{\mathbb{N}}|$.

Proposition 11.7.6 (Cylinders).

1. $\mathsf{C}_{2^{\mathbb{N}}}, \mathsf{C}_{\mathbb{R}}, \mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$ *are cylinders,*
2. $\mathsf{C}_{\mathbb{N}}, \mathsf{PC}_{2^{\mathbb{N}}}, \mathsf{PC}_{\mathbb{R}}, \mathsf{PC}_{\mathbb{N}^{\mathbb{N}}}$ *and* $\mathsf{CC}_{[0,1]}$ *are not cylinders.*

We note that there is also a choice problem $\mathsf{K}_X :\subseteq \mathscr{K}_-(X) \rightrightarrows X, K \mapsto K$ that is called *compact choice*. Unlike the other choice problems we do not just restrict C_X to compact sets here, but we also increase the input information, i.e., the input set K is actually described as a compact set.

With the help of the choice problem C_X for different spaces X we obtain several important Weihrauch degrees. In the following result we indicate how the most important choice problems appear naturally as upper bounds for certain topological properties of the underlying space. We call X *computably countable* if there is a computable surjection $s : \mathbb{N} \to X$, and we say that a computable metric space is *computably σ–compact* if there is a computable sequence $(K_i)_{i\in\mathbb{N}}$ of compact sets $K_i \subseteq X$ such that $X = \bigcup_{i\in\mathbb{N}} K_i$.

Proposition 11.7.7 (Spaces). *Let X be a computable metric space.*

1. $\mathsf{C}_X \leq_{\mathrm{sW}} \mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$ *if X is complete,*
2. $\mathsf{C}_X \leq_{\mathrm{sW}} \mathsf{C}_{\mathbb{R}}$ *if X is computably σ–compact,*

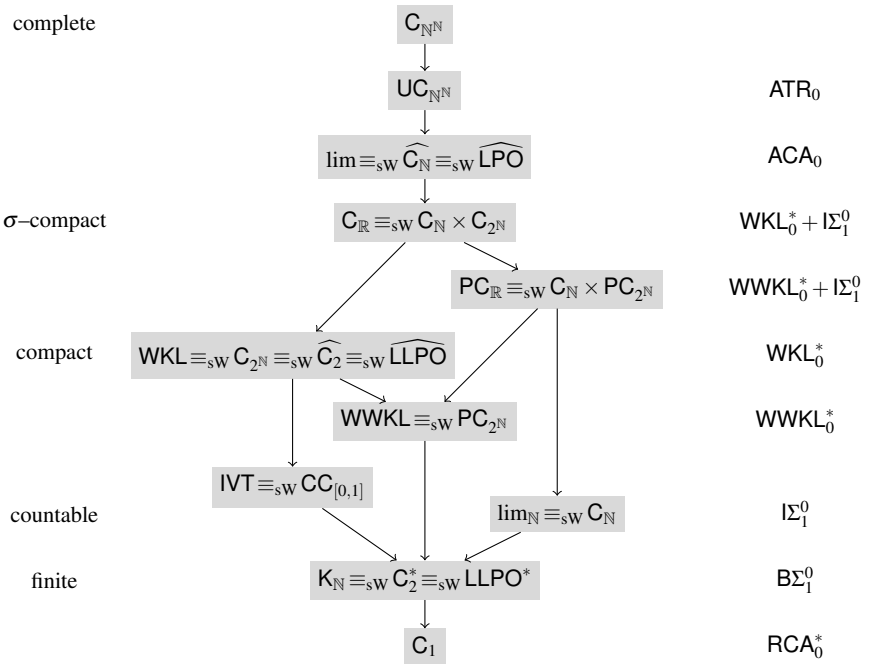


Fig. 11.2 Basic choice problems together with corresponding reverse mathematics systems (see Subsection 11.9.3) and topological properties (every arrow indicates a strong Weihrauch reduction; no additional ordinary Weihrauch reductions hold besides those that follow from transitivity; the arrows in the diagram point in the direction of computations and implicit logical implications and hence in the inverse direction of the corresponding reductions).

3. $\mathsf{C}_X \leq_{\mathrm{sW}} \mathsf{C}_{2^{\mathbb{N}}}$ *if X is computably compact,*
4. $\mathsf{C}_X \leq_{\mathrm{sW}} \mathsf{C}_{\mathbb{N}}$ *if X is computably countable,*
5. $\mathsf{C}_X \leq_{\mathrm{sW}} \mathsf{K}_{\mathbb{N}}$ *if X is finite.*

We will discuss these cones in individual subsections below. The diagram in Figure 11.2 displays several basic choice problems in the Weihrauch lattice. The corresponding systems from reverse mathematics are discussed later in Subsection 11.9.3.

The reader who is mostly interested in classifications of theorems in analysis can continue reading in Section 11.8 from here on. In the remainder of this section we continue discussing systematically choice principles and their properties.

11.7.1 Composition and Non-determinism

In this section we discuss a different perspective on choice that can be seen as a type conversion and that is related to non-determinism. Firstly, we define non-deterministic computability with some advice space R.

Definition 11.7.8 (Non-deterministic computability). Let (X,δ_X), (Y,δ_Y) be represented spaces and $R\subseteq\mathbb{N}^{\mathbb{N}}$. Then $f:\subseteq X\rightrightarrows Y$ is called *non-deterministically computable with advice space R* if there exist computable functions $F:\subseteq\mathbb{N}^{\mathbb{N}}\to\mathbb{N}^{\mathbb{N}}$ and $S:\subseteq\mathbb{N}^{\mathbb{N}}\to\mathbb{S}$ such that $\langle\mathrm{dom}(f\delta_X)\times R\rangle\subseteq\mathrm{dom}(S)$ and for each $p\in\mathrm{dom}(f\delta_X)$:

1. $R_p:=\{r\in R:S\langle p,r\rangle=0\}\neq\emptyset$,
2. $r\in R_p\Longrightarrow\delta_Y F\langle p,r\rangle\in f\delta_X(p)$.

If $R=2^{\mathbb{N}}$, then we say for short that f is *non-deterministically* computable. If in this case we strengthen the first condition to $\mu(R_p)>0$ with the uniform measure μ, then we say that f is *Las Vegas computable*.

Intuitively, the machine can access an arbitrary oracle $r\in R$ besides the input p. Here R_p is the set of successful oracles for input p. On input p together with such successful oracles $r\in R_p$ the computable realizer F produces a correct output. On the other hand, the computable S eventually rejects unsuccessful oracles r, i.e., $S\langle p,r\rangle=1$ for such r. The importance of non-determinism in our context is based on the following observation.

Theorem 11.7.9 (Non-determinism). *$f\leq_{\mathrm{W}}\mathsf{C}_R\iff$ f is non-deterministically computable with advice space R, for every $R\subseteq\mathbb{N}^{\mathbb{N}}$.*

In the case of $R=\mathbb{N}$ it is not too hard to see that we obtain exactly the functions that are computable with finitely many mind changes. We summarize some important classes of functions that can be characterized by an appropriate version of choice.

Corollary 11.7.10 (Notions of computability). *Let f be a problem. Then:*

1. *$f\leq_{\mathrm{W}}\mathsf{C}_{\mathbb{N}}\iff$ f is computable with finitely many mind changes,*
2. *$f\leq_{\mathrm{W}}\mathsf{C}_{2^{\mathbb{N}}}\iff$ f is non-deterministically computable,*
3. *$f\leq_{\mathrm{W}}\mathsf{PC}_{2^{\mathbb{N}}}\iff$ f is Las Vegas computable.*

In particular, all the given properties of f are invariant.

One benefit of characterizing the choice problem with the help of non-deterministic computations is that it is very easy to consider compositions of non-deterministic computations, and hence one obtains a simple proof of the following result that is much harder to prove directly.

Theorem 11.7.11 (Independent choice). *We obtain $\mathsf{C}_R*\mathsf{C}_S\leq_{\mathrm{W}}\mathsf{C}_{R\times S}$ and $\mathsf{PC}_R*\mathsf{PC}_S\leq_{\mathrm{W}}\mathsf{PC}_{R\times S}$ for all $R,S\subseteq\mathbb{N}^{\mathbb{N}}$.*

In the case of positive choice one needs an invocation of Fubini's theorem besides the composition of the two non-deterministic computations. We obtain the following important corollary that, in particular, shows that the notions of computability listed in Corollary 11.7.10 are very natural.

Corollary 11.7.12 (Composition). $\mathsf{C}_{\mathbb{N}}, \mathsf{C}_{2^{\mathbb{N}}}, \mathsf{C}_{\mathbb{R}}, \mathsf{C}_{\mathbb{N}^{\mathbb{N}}}, \mathsf{UC}_{\mathbb{N}^{\mathbb{N}}}, \mathsf{PC}_{2^{\mathbb{N}}}, \mathsf{PC}_{\mathbb{R}}$ *and* $\mathsf{PC}_{\mathbb{N}^{\mathbb{N}}}$ *are closed under compositional product and hence, in particular, idempotent.*

11.7.2 Choice on Natural Numbers

An important equivalence class is the class of choice on natural numbers. We summarize some of its characterizations. In particular, we use the complementary minimum function $\min^{\mathrm{c}} :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}, p \mapsto \min\{n \in \mathbb{N} : (\forall k)\ p(k) \neq n\}$ and the maximum function $\max :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}, p \mapsto \max\{p(n) : n \in \mathbb{N}\}$.

Theorem 11.7.13 (Choice on $\mathbb{N}$). $\mathsf{UC}_{\mathbb{N}} \equiv_{\mathrm{sW}} \mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{sW}} \mathsf{C}_{\mathbb{Q}} \equiv_{\mathrm{sW}} \lim_{\mathbb{N}} \equiv_{\mathrm{sW}} \min^{\mathrm{c}} \equiv_{\mathrm{sW}} \max$.

Here $\mathbb{Q}$ can be endowed with the discrete or the Euclidean topology. The ordinary Weihrauch degree of $\mathsf{C}_{\mathbb{N}}$ has some further members that occasionally appear.

Theorem 11.7.14. $\mathsf{CFC}_{\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{W}} \lim_{\Delta}$ *and* $\mathsf{CFC}_{\mathbb{N}} <_{\mathrm{sW}} \mathsf{C}_{\mathbb{N}} <_{\mathrm{sW}} \lim_{\Delta}$.

Here the strictness results follow from Proposition 11.6.18 since $\#\mathsf{CFC}_{\mathbb{N}} = 1$, $\#\mathsf{C}_{\mathbb{N}} = |\mathbb{N}|$ and $\#\lim_{\Delta} = |\mathbb{N}^{\mathbb{N}}|$.

An important result related to choice on natural numbers shows that it cannot contribute anything to the computation of total fractals if it is applied first (possibly followed by another problem).

Theorem 11.7.15 (Choice elimination). $f \leq_{\mathrm{W}} g * \mathsf{C}_{\mathbb{N}} \Longrightarrow f \leq_{\mathrm{W}} g$*, for every total fractal* f *and every problem* g.

The proof of this theorem is based on the Baire category theorem. In light of Proposition 11.7.5 we obtain the following corollary.

Corollary 11.7.16 (Separations). $\mathsf{CC}_{[0,1]} \not\leq_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$ *and* $\mathsf{PC}_{2^{\mathbb{N}}} \not\leq_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$.

We use the identification $n = \{0, 1, ..., n-1\}$ for all $n \in \mathbb{N}$, and we also consider the finite choice problems C_n. It is clear that $\mathsf{C}_0 \equiv_{\mathrm{W}} \mathbf{0}$ and $\mathsf{C}_1 \equiv_{\mathrm{W}} \mathbf{1}$. The particular case of C_2 is related to LLPO, which is the counterpart of the *lesser limited principle of omniscience* as it is known from constructive analysis.

$$\mathsf{LLPO} :\subseteq 2^{\mathbb{N}} \rightrightarrows \{0,1\}, \mathsf{LLPO}(p) \ni \begin{cases} 0 & \iff (\forall n)\ p(2n) = 0 \\ 1 & \iff (\forall n)\ p(2n+1) = 0 \end{cases}$$

with $\mathrm{dom}(\mathsf{LLPO}) := \{p \in 2^{\mathbb{N}} : p(k) \neq 0 \text{ for at most one } k\}$.

Proposition 11.7.17 (Principles of omniscience). $\mathsf{C}_2 \equiv_{\mathrm{sW}} \mathsf{LLPO} <_{\mathrm{W}} \mathsf{LPO} <_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$.

Likewise, one can define problems MLPO_n that are equivalent to C_n and problems LPO_n that are equivalent to ACC_n. These yield an increasing and a decreasing chain of problems, respectively.

Proposition 11.7.18 (Finite choice). *For every $n > 2$ and every $p \in \mathbb{N}^{\mathbb{N}}$ we obtain* $\mathsf{ACC}_{\mathbb{N}} <^p_{\mathrm{W}} \mathsf{ACC}_{n+1} <^p_{\mathrm{W}} \mathsf{ACC}_n <^p_{\mathrm{W}} \mathsf{ACC}_2 = \mathsf{C}_2 <^p_{\mathrm{W}} \mathsf{C}_n <^p_{\mathrm{W}} \mathsf{C}_{n+1} <^p_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$.

The mere fact that $\mathsf{C}_{n+1} \not\leq_{\mathrm{W}} \mathsf{C}_n$ holds follows since $\mathrm{mind}(\mathsf{C}_n) = n-1$ for all $n \geq 1$. While the power of choice increases with the finite cardinality, we can compensate cardinality by sufficiently many parallel copies of C_2, as the following result shows.

Theorem 11.7.19 (Cardinality versus products). $\mathsf{C}_{n+1} \leq_{\mathrm{sW}} \mathsf{C}_2^n$ *for all $n \in \mathbb{N}$.*

It is easy to see that also $\mathsf{C}_2^n \leq_{\mathrm{sW}} \mathsf{C}_{2^n}$ holds. This implies the second equivalence in the following result.

Proposition 11.7.20 (Compact choice). $\mathsf{K}_{\mathbb{N}} \equiv_{\mathrm{sW}} \mathsf{C}_2^* \equiv_{\mathrm{sW}} \mathsf{C}_n^*$ *for all $n \geq 2$.*

We note that $\mathsf{C}_2^* <_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$, since $\mathsf{C}_{\mathbb{N}}$ is a fractal by Proposition 11.7.5 and hence countably irreducible.

Corollary 11.7.21 (Compact versus closed choice). $\mathsf{K}_{\mathbb{N}} <_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$.

We use the minimum function $\min :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}, p \mapsto \min\{p(n) : n \in \mathbb{N}\}$ in order to express the last result of this subsection.

Proposition 11.7.22 (Minimum). $\mathsf{LPO}^* \equiv_{\mathrm{sW}} \min$.

11.7.3 Choice on the Cantor Space

Choice on the Cantor space $2^{\mathbb{N}}$ is closely related to weak Kőnig's lemma, which states that every infinite binary tree $T \subseteq 2^*$ has an infinite path $p \in 2^{\mathbb{N}}$ (formally, a binary tree is a subset of 2^* closed downward with respect to the partial order induced by the prefix relation). By Tr we denote the set of binary trees $T \subseteq 2^*$ (represented by their characteristic functions $\chi_T : 2^* \to 2$), and by $[T]$ we denote the set of infinite paths $p \in 2^{\mathbb{N}}$ of T. Now we formalize weak Kőnig's lemma as the problem $\mathsf{WKL} :\subseteq \mathrm{Tr} \rightrightarrows 2^{\mathbb{N}}, T \mapsto [T]$, where $\mathrm{dom}(\mathsf{WKL})$ consists of all infinite binary trees.

It is well known that the map $[.] : \mathrm{Tr} \to \mathscr{A}_-(2^{\mathbb{N}}), T \mapsto [T]$ is computable, surjective and has a computable multi-valued right-inverse. This yields the first equivalence in the following theorem.

Theorem 11.7.23 (Choice on the Cantor space). *For every rich computably compact computable metric space X:* $\mathsf{WKL} \equiv_{\mathrm{sW}} \mathsf{C}_{2^{\mathbb{N}}} \equiv_{\mathrm{sW}} \mathsf{C}_X \equiv_{\mathrm{sW}} \widehat{\mathsf{C}_2} \equiv_{\mathrm{sW}} \widehat{\mathsf{LLPO}}$.

In particular, $\mathsf{C}_{2^{\mathbb{N}}}$ is parallelizable, and the problem of finding a path in a binary tree can be reduced to countably many binary choices, i.e., $\mathsf{WKL} \leq_{\mathrm{W}} \widehat{\mathsf{C}_2}$. In the following corollary we list the choice problem for some important examples of rich computably compact computable metric spaces.

Corollary 11.7.24. $\mathsf{C}_{2^{\mathbb{N}}} \equiv_{\mathrm{sW}} \mathsf{C}_{[0,1]^{\mathbb{N}}} \equiv_{\mathrm{sW}} \mathsf{C}_{[0,1]^n}$ *for all* $n \geq 1$.

Similarly as for choice on natural numbers there is a choice elimination result for $\mathsf{C}_{2^{\mathbb{N}}}$. This result can be proved using compactness properties.

Theorem 11.7.25 (Choice elimination). $f \leq_{\mathrm{W}} \mathsf{C}_{2^{\mathbb{N}}} * g \Longrightarrow f \leq_{\mathrm{W}} g$ *for every single-valued problem* $f : X \to Y$ *with a computable metric space* Y *and every* g.

This result can also be generalized to admissibly represented spaces Y. We obtain the following important special case.

Corollary 11.7.26 (Single-valuedness). $f \leq_{\mathrm{W}} \mathsf{C}_{2^{\mathbb{N}}} \Longrightarrow f$ *computable, for all single-valued problems* $f : X \to Y$ *with a computable metric space* Y.

In particular this applies to $f = \mathsf{UC}_{2^{\mathbb{N}}}$. Since $\lim_{\mathbb{N}}$ is a single-valued problem in the equivalence class of $\mathsf{C}_{\mathbb{N}}$, we also get the following conclusion.

Corollary 11.7.27 (Separation). $\mathsf{C}_{\mathbb{N}} \not\leq_{\mathrm{W}} \mathsf{C}_{2^{\mathbb{N}}}$.

The so-called weak weak Kőnig's lemma WWKL is WKL restricted to trees T such that $\mu([T]) > 0$. It is easy to see that it is equivalent to $\mathsf{PC}_{2^{\mathbb{N}}}$.

Theorem 11.7.28 (Positive choice on the Cantor space).
$\mathsf{WWKL} \equiv_{\mathrm{sW}} \mathsf{PC}_{2^{\mathbb{N}}} \equiv_{\mathrm{sW}} \mathsf{PC}_{[0,1]}$.

One can use a result of Jockusch and Soare [64, Theorem 5.3] that essentially shows that WKL cannot be computed with an advice set of positive measure, in order to separate $\mathsf{PC}_{2^{\mathbb{N}}}$ and $\mathsf{C}_{2^{\mathbb{N}}}$.

Proposition 11.7.29 (Positive choice versus choice). $\mathsf{PC}_{2^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{C}_{2^{\mathbb{N}}}$.

Since $\mathsf{C}_2 \leq_{\mathrm{W}} \mathsf{PC}_{2^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{C}_{2^{\mathbb{N}}} \equiv_{\mathrm{W}} \widehat{\mathsf{C}_2}$, it is clear that $\mathsf{PC}_{2^{\mathbb{N}}}$ is not parallelizable.

Corollary 11.7.30 (Parallelizability). $\mathsf{PC}_{2^{\mathbb{N}}}$ *is not parallelizable and we obtain* $\widehat{\mathsf{PC}_{2^{\mathbb{N}}}} \equiv_{\mathrm{sW}} \mathsf{C}_{2^{\mathbb{N}}}$.

Also quantitative versions of WWKL have been considered, and by ε-WWKL for $\varepsilon \in (0,1)$ we denote WWKL restricted to trees with $\mu([T]) > \varepsilon$.

Theorem 11.7.31 (Quantitative WWKL). ε-$\mathsf{WWKL} \leq_{\mathrm{W}} \delta$-$\mathsf{WWKL} \iff \varepsilon \geq \delta$.

We continue with the discussion of further special versions of choice related to $\mathsf{C}_{2^{\mathbb{N}}}$. It is not obvious at all that connected choice $\mathsf{CC}_{[0,1]^n}$ is in the same equivalence class as $\mathsf{C}_{2^{\mathbb{N}}}$ for dimension $n \geq 2$.

Theorem 11.7.32 (Connected choice). $\mathsf{CC}_{[0,1]^n} \equiv_{\mathrm{sW}} \mathsf{PWCC}_{[0,1]^{n+1}} \equiv_{\mathrm{sW}} \mathsf{C}_{2^{\mathbb{N}}}$ *for* $n \geq 2$.

The map $A \mapsto (A \times [0,1] \times \{0\}) \cup (A \times A \times [0,1]) \cup ([0,1] \times A \times \{1\})$ shows that one can map each closed subset $A \subseteq [0,1]$ to a pathwise connected closed subset $B \subseteq [0,1]^3$, and given a point in the latter set one can reconstruct a point in the former set. This proves the previous statement on $\mathsf{PWCC}_{[0,1]^n}$ and $\mathsf{CC}_{[0,1]^n}$ for $n \geq 3$. Only the two-dimensional case needs a more sophisticated argument, and in the case of $\mathsf{PWCC}_{[0,1]^2}$ the Weihrauch degree is not known.

Problem 11.7.33 (Pathwise connected choice). Does $\mathsf{PWCC}_{[0,1]^2} \equiv_{\mathrm{W}} \mathsf{C}_{2^{\mathbb{N}}}$ hold?

The one-dimensional case of connected choice yields the degree of the intermediate value theorem.

Theorem 11.7.34 (Intermediate value theorem). $\mathsf{CC}_{[0,1]} \equiv_{\mathrm{sW}} \mathsf{IVT}$.

We mention a fact that was already stated in Example 11.4.3.

Theorem 11.7.35 (Idempotency). $\mathsf{CC}_{[0,1]}$ *is not idempotent.*

While connected choice is very stable with respect to the dimension of the space, this is not so for convex choice as the following result shows.

Theorem 11.7.36 (Convex choice). $\mathsf{XC}_{[0,1]^n} <_{\mathrm{W}} \mathsf{XC}_{[0,1]^{n+1}}$ *for all* $n \in \mathbb{N}$.

Convex choice is not closed under composition, as the following result shows.

Theorem 11.7.37 (Composition of convex choice). $\mathsf{XC}_{[0,1]^n}$ *is not closed under compositional product* $*$ *and* $\mathsf{XC}_{[0,1]} * \mathsf{XC}_{[0,1]} \not\leq_{\mathrm{W}} \mathsf{XC}_{[0,1]^n}$ *for all* $n \geq 1$.

We mention that compact choice K_X does not lead to anything new on rich computable metric spaces.

Theorem 11.7.38 (Compact choice). $\mathsf{K}_X \equiv_{\mathrm{sW}} \mathsf{C}_{2^{\mathbb{N}}}$ *for all rich computable metric spaces* X.

In particular, this implies $\mathsf{K}_{2^{\mathbb{N}}} \equiv_{\mathrm{sW}} \mathsf{K}_{\mathbb{R}} \equiv_{\mathrm{sW}} \mathsf{K}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{sW}} \mathsf{C}_{2^{\mathbb{N}}}$. We close this section by mentioning that $\mathsf{CC}_{[0,1]}$ and $\mathsf{PC}_{2^{\mathbb{N}}}$ are both upper bounds of $\mathsf{K}_{\mathbb{N}}$.

Proposition 11.7.39 (Upper bound of compact choice). $\mathsf{K}_{\mathbb{N}} \leq_{\mathrm{W}} \mathsf{CC}_{[0,1]} \sqcap \mathsf{PC}_{2^{\mathbb{N}}}$.

11.7.4 Choice on Euclidean Space

In this section we discuss $\mathsf{C}_{\mathbb{R}}$ and related problems. The basic observation is that $\mathsf{C}_{\mathbb{R}}$ can be described with the help of $\mathsf{C}_{2^{\mathbb{N}}}$ and $\mathsf{C}_{\mathbb{N}}$ in several different ways.

Theorem 11.7.40 (Choice on Euclidean space). $\mathsf{C}_{\mathbb{R}} \equiv_{\mathrm{sW}} \mathsf{C}_{\mathbb{R}^n} \equiv_{\mathrm{sW}} \mathsf{C}_{2^{\mathbb{N}}\times\mathbb{N}} \equiv_{\mathrm{sW}} \mathsf{C}_{2^{\mathbb{N}}} \times \mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{sW}} \mathsf{C}_{2^{\mathbb{N}}} \star \mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{sW}} \mathsf{C}_{\mathbb{N}} \star \mathsf{C}_{2^{\mathbb{N}}}$ *for all* $n \geq 1$.

The results regarding $\star$ follow with the help of Theorem 11.7.11. We have deliberately used the symbol $\star$ and not $*$, since the degrees with $\star$ are cylinders and hence we obtain strong equivalences. Theorem 11.7.40 shows that Theorem 11.7.25 is applicable to $\mathsf{C}_{\mathbb{R}}$, and we obtain the following conclusion.

Corollary 11.7.41 (Single-valuedness). $f \leq_{\mathrm{W}} \mathsf{C}_{\mathbb{R}} \Longrightarrow f \leq_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$, *for all single-valued problems* $f : X \to Y$ *with a computable metric space* Y.

This result applies in particular to $\mathsf{UC}_{\mathbb{R}}$ and implies $\mathsf{UC}_{\mathbb{R}} \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$. Now we discuss an important upper bound on $\mathsf{C}_{\mathbb{R}}$. The low basis theorem of Jockusch and Soare states that every computable infinite binary tree has a low path. We recall that $p \in \mathbb{N}^{\mathbb{N}}$ is called *low* if $p' \leq_{\mathrm{T}} \emptyset'$ holds, i.e., if the halting problem relative to p is not more difficult than the ordinary halting problem. Lowness is represented by the problem $\mathsf{L} := \mathsf{J}^{-1} \circ \lim$ since p is low if and only if there is a computable q such that $p = \mathsf{L}(q)$. It is clear that $\mathsf{L} <_{\mathrm{W}} \lim$ (since J^{-1} is computable and not every limit computable p is low). The following result can be seen as a uniform version of the low basis theorem.

Theorem 11.7.42 (Uniform low basis theorem). $\mathsf{C}_{\mathbb{R}} <_{\mathrm{sW}} \mathsf{L}$.

The strictness follows for instance from Corollary 11.7.41 since L is single-valued. We mention an interesting algebraic example of how infima and suprema of the degrees of $\mathsf{C}_{\mathbb{N}}$ and $\mathsf{C}_{2^{\mathbb{N}}}$ interact.

Example 11.7.43. We obtain

1. $(\mathsf{C}_{2^{\mathbb{N}}} \sqcap \mathsf{C}_{\mathbb{N}}) * (\mathsf{C}_{2^{\mathbb{N}}} \sqcup \mathsf{C}_{\mathbb{N}}) \equiv_{\mathrm{W}} (\mathsf{C}_{2^{\mathbb{N}}} \sqcap \mathsf{C}_{\mathbb{N}}) \times (\mathsf{C}_{2^{\mathbb{N}}} \sqcup \mathsf{C}_{\mathbb{N}}) \equiv_{\mathrm{W}} \mathsf{C}_{2^{\mathbb{N}}} \sqcup \mathsf{C}_{\mathbb{N}}$.
2. $(\mathsf{C}_{2^{\mathbb{N}}} \sqcup \mathsf{C}_{\mathbb{N}}) * (\mathsf{C}_{2^{\mathbb{N}}} \sqcap \mathsf{C}_{\mathbb{N}}) \equiv_{\mathrm{W}} \mathsf{C}_{2^{\mathbb{N}}} * \mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{C}_{2^{\mathbb{N}}} \times \mathsf{C}_{\mathbb{N}}$.

We note that $\mathsf{C}_{2^{\mathbb{N}}} \sqcup \mathsf{C}_{\mathbb{N}} <_{\mathrm{W}} \mathsf{C}_{2^{\mathbb{N}}} \times \mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{R}}$ since the right-hand degree is join-irreducible and since $\mathsf{C}_{2^{\mathbb{N}}}$ and $\mathsf{C}_{\mathbb{N}}$ are incomparable. We formulate a counterpart of Theorem 11.7.40 for $\mathsf{PC}_{\mathbb{R}}$.

Theorem 11.7.44 (Positive choice on Euclidean space). $\mathsf{PC}_{\mathbb{R}} \equiv_{\mathrm{sW}} \mathsf{PC}_{2^{\mathbb{N}}\times\mathbb{N}} \equiv_{\mathrm{sW}} \mathsf{PC}_{2^{\mathbb{N}}} \times \mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{PC}_{2^{\mathbb{N}}} * \mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}} * \mathsf{PC}_{2^{\mathbb{N}}}$ *for all* $n \geq 1$.

We mention that in this case we cannot simply replace $*$ by $\star$ and $\equiv_{\mathrm{W}}$ by $\equiv_{\mathrm{sW}}$, since $\mathsf{PC}_{\mathbb{R}}$ is not a cylinder. We note that $\mathsf{C}_{\mathbb{N}} <_{\mathrm{W}} \mathsf{C}_{\mathbb{R}} <_{\mathrm{W}} \lim \equiv_{\mathrm{W}} \widehat{\mathsf{C}_{\mathbb{N}}}$ implies the following.

Corollary 11.7.45 (Parallelizability). $\mathsf{C}_{\mathbb{R}}$ *and* $\mathsf{PC}_{\mathbb{R}}$ *are not parallelizable, and we obtain* $\widehat{\mathsf{PC}_{\mathbb{R}}} \equiv_{\mathrm{sW}} \widehat{\mathsf{C}_{\mathbb{R}}} \equiv_{\mathrm{sW}} \lim$.

11.7.5 Choice on the Baire Space

Choice on the Baire space is the upper bound of all choice problems of complete computable metric spaces. In fact, we obtain the following.

Theorem 11.7.46 (Non-σ-compact spaces). $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}} \equiv^{p}_{\mathrm{W}} \mathsf{C}_X$ *for some oracle* $p \in \mathbb{N}^{\mathbb{N}}$ *if* X *is a separable complete metric space that is not* σ*–compact.*

We list a number of choice problems that fall into the equivalence class of $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$. We assume that these spaces are represented as computable metric spaces in the standard way.

Theorem 11.7.47 (Baire space). $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{sW}} \mathsf{C}_{\mathbb{R}^{\mathbb{N}}} \equiv_{\mathrm{sW}} \mathsf{C}_{\mathbb{R}\setminus\mathbb{Q}} \equiv_{\mathrm{sW}} \mathsf{C}_{\ell_p} \equiv_{\mathrm{sW}} \mathsf{C}_{\mathscr{C}[0,1]}$ *for all computable* $p \geq 1$.

Also the single-valued problems below $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$ have a very natural characterization.

Theorem 11.7.48 (Single-valuedness). $f \leq_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^{\mathbb{N}}} \iff f$ *is effectively Borel measurable, for* $f : X \to Y$ *on complete computable metric spaces.*

Similarly to Theorem 11.6.5 this result can be relativized. We mention that it is easy to see that $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$ is parallelizable.

Proposition 11.7.49 (Parallelizability). $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$ *is strongly parallelizable.*

We briefly mention $\mathsf{UC}_{\mathbb{N}^{\mathbb{N}}}$, the unique version of choice on the Baire space. It is easy to see that $\lim <_{\mathrm{W}} \mathsf{UC}_{\mathbb{N}^{\mathbb{N}}}$ holds. It follows from a basis theorem of Kreisel that $\mathsf{UC}_{\mathbb{N}^{\mathbb{N}}}$ is strictly weaker than $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$.

Proposition 11.7.50 (Unique choice). $\lim^{(n)} <_{\mathrm{W}} \mathsf{UC}_{\mathbb{N}^{\mathbb{N}}} <_{\mathrm{W}} \mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$ *for all* $n \in \mathbb{N}$.

We close with the following characterization of positive choice on the Baire space.

Theorem 11.7.51 (Positive choice). $\mathsf{PC}_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{sW}} \mathsf{PC}_{\mathbb{R}}$.

11.7.6 Jumps of Choice

In order to characterize the jump of choice we need the *cluster point problem* $\mathsf{CL}_X :\subseteq X^{\mathbb{N}} \rightrightarrows X, (x_n)_{n\in\mathbb{N}} \mapsto \{x \in X : x \text{ is a cluster point of } (x_n)_{n\in\mathbb{N}}\}$. This problem fully characterizes the jump of C_X on computable metric spaces X. If we restrict CL_X to such sequences $(x_n)_{n\in\mathbb{N}}$ whose range $\{x_n : n \in \mathbb{N}\}$ has a compact closure, then we denote it by $\mathsf{BWT}_X :\subseteq X^{\mathbb{N}} \rightrightarrows X$ since it can be seen as a problem that realizes the Bolzano-Weierstraß theorem.

Theorem 11.7.52 (Jump of choice). $\mathsf{C}'_X \equiv_{\mathrm{sW}} \mathsf{CL}_X$ *and* $\mathsf{K}'_X \equiv_{\mathrm{sW}} \mathsf{BWT}_X$ *for all computable metric spaces* X.

$\mathsf{BWT}_2 = \mathsf{CL}_2$ is also known as the *infinite pigeonhole problem*. Many properties of problems can be transferred to jumps. However, this is often not so for properties that involve compositional products. We recall that by $f^{[n]}$ we denote the n–fold compositional product of f with itself and by $f^{(n)}$ the n–fold jump.

Theorem 11.7.53 (Composition). *We obtain:*

1. $\mathsf{C}'_{\mathbb{N}} * \mathsf{C}'_{\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{C}'_{\mathbb{N}}$.
2. $\mathsf{C}'_{2^{\mathbb{N}}} * \mathsf{C}'_{2^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{C}''_{2^{\mathbb{N}}}$ *and more generally* $\mathsf{C}'^{[n]}_{2^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{C}^{(n)}_{2^{\mathbb{N}}}$ *for all* $n \geq 1$.
3. $\mathsf{PC}'_{2^{\mathbb{N}}} * \mathsf{PC}'_{2^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{PC}'_{\mathbb{R}} * \mathsf{PC}'_{\mathbb{R}} \equiv_{\mathrm{W}} \mathsf{PC}'_{\mathbb{R}}$.

It is perhaps surprising that compositions behave very differently in the probabilistic case and in the non-probabilistic case. The difference between $\mathsf{C}_{2^{\mathbb{N}}}$ and $\mathsf{PC}_{2^{\mathbb{N}}}$ is also underlined by the third statement in the following result that strengthens the negative statement of Proposition 11.7.29.

Theorem 11.7.54 (Separations). *We obtain for all* $n \in \mathbb{N}$*:*

1. $\mathsf{C}_2^{(n+1)} \not\leq_{\mathrm{W}} \lim^{(n)}$.
2. $\mathsf{LPO}^{(n)} \not\leq_{\mathrm{W}} \mathsf{C}^{(n)}_{2^{\mathbb{N}}}$.
3. $\mathsf{C}_{2^{\mathbb{N}}} \not\leq_{\mathrm{W}} \mathsf{PC}^{(n)}_{2^{\mathbb{N}}}$.

It follows from the first statement that $\mathsf{C}_2 \leq_{\mathrm{W}} \mathsf{PC}_{2^{\mathbb{N}}} \leq_{\mathrm{W}} \mathsf{C}_{2^{\mathbb{N}}}$ all climb up the Borel hierarchy one step with every jump. This statement even holds relative to any oracle, and hence $\mathsf{C}_2^{(n)}$ is not Σ^0_{n+1}–measurable. We obtain the following alternating hierarchies.

Theorem 11.7.55 (Alternating hierarchies). *For all* $n \in \mathbb{N}$ *we obtain*

1. $\mathsf{C}_2^{(n)} <_{\mathrm{W}} \mathsf{LPO}^{(n)} <_{\mathrm{W}} \mathsf{C}_2^{(n+1)}$,
2. $\mathsf{K}_{\mathbb{N}}^{(n)} <_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}^{(n)} <_{\mathrm{W}} \mathsf{K}_{\mathbb{N}}^{(n+1)}$,
3. $\mathsf{C}^{(n)}_{2^{\mathbb{N}}} <_{\mathrm{W}} \lim^{(n)} <_{\mathrm{W}} \mathsf{C}^{(n+1)}_{2^{\mathbb{N}}}$.

Analogous statements hold with $\leq_{\mathrm{sW}}$ *in place of* $\leq_{\mathrm{W}}$.

Choice on the Baire space is an example of a choice problem that is stable under jump.

Theorem 11.7.56 (Baire space). $\mathsf{C}'_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{sW}} \mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$ *and* $\mathsf{UC}'_{\mathbb{N}^{\mathbb{N}}} \equiv_{\mathrm{sW}} \mathsf{UC}_{\mathbb{N}^{\mathbb{N}}}$.

The following example shows that a straightforward jump inversion theorem does not hold in the Weihrauch lattice.

Example 11.7.57. $\lim <_{\mathrm{W}} \mathsf{C}'_{2^{\mathbb{N}}} \sqcup \mathsf{C}'_{\mathbb{N}}$, but there is no problem f with $f' \equiv_{\mathrm{W}} \mathsf{C}'_{2^{\mathbb{N}}} \sqcup \mathsf{C}'_{\mathbb{N}}$.

The latter holds since f' is join-irreducible and $\mathsf{C}'_{2^{\mathbb{N}}}$ and $\mathsf{C}'_{\mathbb{N}}$ are incomparable. While Theorem 11.7.54 shows that $\mathsf{C}_{2^{\mathbb{N}}}$ has no jump of positive choice as upper bound, this is different for $\mathsf{CC}_{[0,1]}$ as the following result shows.

Proposition 11.7.58 (Upper bounds). $\mathsf{CC}_{[0,1]} \sqcup \mathsf{C}_{\mathbb{N}} \leq_{\mathrm{W}} \mathsf{PC}'_{2^{\mathbb{N}}} \sqcap \mathsf{C}'_{\mathbb{N}}$.

This result is contrasted by $\mathsf{CC}_{[0,1]} \not\leq_{\mathrm{W}} \mathsf{PC}_{2^{\mathbb{N}}} \sqcup \mathsf{C}_{\mathbb{N}}$, which holds because $\mathsf{CC}_{[0,1]}$ is not reducible to any of the problems on the right-hand side and since it is join-irreducible by Propositions 11.7.5 and 11.4.11.

For $\frac{1}{2}$-WWKL we can improve the statement that follows from Corollary 11.7.26 by the following result, which can be proved with a majority vote argument.

Theorem 11.7.59 (Single-valuedness). $f \leq_{\mathrm{W}} \frac{1}{2}\text{-}\mathsf{WWKL}^{(n)} \Longrightarrow$ *f computable, for all single-valued problems $f : X \to Y$ with a computable metric space Y and $n \in \mathbb{N}$.*

We note that $\frac{1}{2}$-WWKL cannot be replaced by WWKL in this result, since $\lim_{\mathbb{N}} \leq_{\mathrm{W}} \mathsf{WWKL}'$ holds as a consequence of Proposition 11.7.58.

11.7.7 All-or-Unique Choice

We briefly discuss all-or-unique choice in this section. The problem $\mathsf{AoUC}_{[0,1]}$ is located between LLPO and LPO and related to *robust division*, which is defined as the problem $\mathsf{RDIV} : [0,1] \times [0,1] \rightrightarrows [0,1]$ with $\mathsf{RDIV}(x,y) := \{\frac{x}{\max(x,y)}\}$ if $y \neq 0$ and $\mathsf{RDIV}(x,y) := [0,1]$ otherwise. We now obtain the following characterization.

Proposition 11.7.60 (All-or-unique choice). $\mathsf{C}_2 <_{\mathrm{W}} \mathsf{AoUC}_{[0,1]} \equiv_{\mathrm{sW}} \mathsf{RDIV} <_{\mathrm{W}} \mathsf{LPO}$.

In some respects $\mathsf{AoUC}_{[0,1]}$ is closer to C_2 than to LPO, at least with respect to the following upper bounds.

Theorem 11.7.61 (Upper bound). $\mathsf{AoUC}_{[0,1]} \leq_{\mathrm{W}} \mathsf{CC}_{[0,1]} \sqcap \mathsf{PC}_{2^{\mathbb{N}}}$.

In the diagram in Figure 11.2 $\mathsf{AoUC}_{[0,1]}$ would be in a similar position to $\mathsf{K}_{\mathbb{N}}$, however it is incomparable to $\mathsf{K}_{\mathbb{N}}$ since $\mathsf{AoUC}_{[0,1]}$ is countably irreducible and $\mathsf{K}_{\mathbb{N}} \not\leq_{\mathrm{W}} \mathsf{LPO}$. We continue with a number of separation results that involve $\mathsf{AoUC}_{[0,1]}$.

Theorem 11.7.62 (Separation).

1. $\mathsf{XC}_{[0,1]} * \mathsf{AoUC}_{[0,1]} \not\leq_{\mathrm{W}} \mathsf{XC}_{[0,1]^n}$ *for all* $n \in \mathbb{N}$.
2. $\mathsf{C}_2 * \mathsf{AoUC}_{[0,1]} \not\leq_{\mathrm{W}} \mathsf{AoUC}^*_{[0,1]}$.
3. $\mathsf{C}_2 \times \mathsf{AoUC}_{[0,1]} \not\leq_{\mathrm{W}} \mathsf{CC}_{[0,1]}$.

These separation results have a number of interesting consequences. The first statement implies Theorem 11.7.37, and the third statement implies Theorem 11.7.35. Since $\mathsf{C}_2 \leq_{\mathrm{W}} \mathsf{AoUC}_{[0,1]}$ we can conclude the following from the second statement.

Corollary 11.7.63 (Composition). $\mathsf{AoUC}^*_{[0,1]}$ *is not closed under compositional product.*

Surprisingly, a composition of $\mathsf{AoUC}^*_{[0,1]}$ with itself yields a new problem that is closed under compositional product.

Theorem 11.7.64 (Double composition). *We obtain:*
$\mathsf{AoUC}^*_{[0,1]} * \mathsf{AoUC}^*_{[0,1]} \equiv_{\mathrm{W}} \mathsf{AoUC}^*_{[0,1]} * \mathsf{AoUC}^*_{[0,1]} * \mathsf{AoUC}^*_{[0,1]}$.

Bibliographic Remarks

The study of choice problems in the Weihrauch lattice has been started by Gherardi and Marcone [46] and Brattka and Gherardi [18, 17]. Indirectly, Weihrauch [115] already studied choice problems in the form of versions of MLPO and LPO; Propositions 11.7.17 and 11.7.18 are due to him. Theorem 11.7.19 is due to Pauly [91]. Non-deterministically computable functions have been introduced to computable analysis by Ziegler [119, 118], and the relation to choice problems has been established by Brattka, de Brecht and Pauly [14], who also started to study choice problems more systematically. Many results in this section are due to them. The study of positive choice was initiated by Brattka and Pauly [28] and more systematically continued by Brattka, Gherardi and Hölzl [19]. This subject was independently studied by Dorais, Dzhafarov, Hirst, Mileti and Shafer [38], who also introduced the quantitative version of WWKL and proved Theorem 11.7.31. Connected choice was mostly studied by Brattka, Le Roux, Miller and Pauly [27] and convex choice by Le Roux and Pauly [79], who also proved Theorem 11.7.15. The study of jumps of choice is due to Brattka, Gherardi and Marcone [21], and the statement on K_X in Theorem 11.7.52 is due to Brattka, Cettolo, Gherardi, Marcone and Schröder [15]. The statement on positive choice in Theorem 11.7.53 is due to Bienvenu and Kuyper [3]. Pauly started the study of all-or-unique choice [91, 93], and some of the separation results in this regard are due to Kihara and Pauly [69]. Theorem 11.7.37 is due to Kihara [68].

11.8 Classifications

In this section we present results on the classification of theorems. Most of these theorems originate from analysis. We interpret theorems as problems as explained after Definition 11.1.4. For many theorems one can derive upper bounds using the following observation.

Theorem 11.8.1 (Upper bounds). *Let X,Y be represented spaces and $A \subseteq X \times Y$ co-c.e. closed. If $(\forall x \in X)(\exists y \in Y)\ (x,y) \in A$ holds, then the corresponding problem $F : X \rightrightarrows Y, x \mapsto \{y \in Y : (x,y) \in A\}$ satisfies $F \leq_{\mathrm{W}} \mathsf{C}_Y$.*

In combination with Proposition 11.7.7 one can thus derive upper bounds on theorems by exploiting topological properties of Y. We essentially group our classifications according to related choice problems.

The equivalence class of choice on natural numbers contains many theorems that are typically proved with the help of the Baire category theorem.

Theorem 11.8.2 (Choice on the natural numbers). *The following are all Weihrauch equivalent to each other:*

1. *Choice on natural numbers $\mathsf{C}_{\mathbb{N}}$.*
2. *The Baire category theorem BCT_1.*
3. *Banach's inverse mapping theorem $\mathsf{BIM}_{\ell_2,\ell_2}$.*
4. *The open mapping theorem for ℓ_2.*
5. *The closed graph theorem for ℓ_2.*
6. *The uniform boundedness theorem on non-singleton computable Banach spaces.*
7. *The Lebesgue covering lemma for $[0,1]$.*

8. The partial identity from continuous functions to analytic functions.

In most cases these theorems are interpreted as problems in a straightforward way. We only provide some examples and refer the reader to the references for exact definitions. For instance, Banach's inverse mapping theorem on computable Banach spaces X,Y is formalized as $\mathsf{BIM}_{X,Y} :\subseteq \mathscr{C}(X,Y) \to \mathscr{C}(Y,X), T \mapsto T^{-1}$, restricted to bijective, linear, bounded T. We always obtain $\mathsf{BIM}_{X,Y} \leq_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$, and in the case of $X = Y = \ell_2$ the theorem actually attains the maximal complexity. For finite-dimensional X,Y it is, however, computable. Similar remarks apply in the cases of the open mapping theorem and the closed graph theorem. The Baire category theorem and the uniform boundedness theorem are even equivalent to $\mathsf{C}_{\mathbb{N}}$ for all complete computable metric spaces and non-singleton computable Banach spaces, respectively.

In the case of some theorems it can happen that one logical formulation of the theorem and the contrapositive formulation carry different computational content. In such a situation it might not always be clear which form is more natural, and perhaps both forms have applications. Such an example is the Baire category theorem, which we can formalize at least in two ways. By A° we denote the *interior* of the set A.

1. $\mathsf{BCT}_0 :\subseteq \mathscr{A}_-(X)^{\mathbb{N}} \rightrightarrows X, (A_n)_{n\in\mathbb{N}} \mapsto \{x \in X : x \notin \bigcup_{n=0}^{\infty} A_n\}$, with $\mathrm{dom}(\mathsf{BCT}_0) := \{(A_n)_{n\in\mathbb{N}} : A_n^\circ = \emptyset\}$.
2. $\mathsf{BCT}_1 :\subseteq \mathscr{A}_-(X)^{\mathbb{N}} \rightrightarrows \mathbb{N}, (A_n)_{n\in\mathbb{N}} \mapsto \{n \in \mathbb{N} : A_n^\circ \neq \emptyset\}$, with $\mathrm{dom}(\mathsf{BCT}_1) := \{(A_n)_{n\in\mathbb{N}} : X = \bigcup_{n=0}^{\infty} A_n\}$.

While BCT_1 is in the equivalence class of $\mathsf{C}_{\mathbb{N}}$, it is easy to see that BCT_0 is computable. Nevertheless the jump of BCT_0 has interesting applications that we mention below. Similarly to the Baire category theorem, also the Heine-Borel covering theorem can be formalized in at least two ways. Here $\mathscr{O}(X)$ denotes the set of open subsets of X seen as the complements of the elements of $\mathscr{A}_-(X)$, i.e., every open set is represented by an enumeration of basic open balls whose union coincides with the set.

1. $\mathsf{HBC}_0 :\subseteq \mathscr{O}([0,1])^{\mathbb{N}} \rightrightarrows \mathbb{N}, (U_n)_{n\in\mathbb{N}} \mapsto \{k \in \mathbb{N} : [0,1] \subseteq \bigcup_{n=0}^{k} U_n\}$, with $\mathrm{dom}(\mathsf{HBC}_0) := \{(U_n)_{n\in\mathbb{N}} : [0,1] \subseteq \bigcup_{n=0}^{\infty} U_n\}$.
2. $\mathsf{HBC}_1 :\subseteq \mathscr{O}([0,1])^{\mathbb{N}} \rightrightarrows [0,1], (U_n)_{n\in\mathbb{N}} \mapsto \{x \in [0,1] : x \notin \bigcup_{n=0}^{\infty} U_n\}$, with $\mathrm{dom}(\mathsf{HBC}_1) := \{(U_n)_{n\in\mathbb{N}} : (\forall k)\ [0,1] \not\subseteq \bigcup_{n=0}^{k} U_n\}$.

Once again, it is easy to see that HBC_0 is computable, and HBC_1 is in the equivalence class of choice on the Cantor space.

Theorem 11.8.3 (Choice on the Cantor space). *The following are all strongly Weihrauch equivalent to each other:*

1. *Choice on the Cantor space* $\mathsf{C}_{2^{\mathbb{N}}}$.
2. *Weak Kőnig's lemma* WKL.
3. *The Hahn-Banach theorem.*
4. *The Heine-Borel covering theorem* HBC_1.
5. *The theorem of the maximum* MAX.

6. *The Brouwer fixed-point theorem* BFT_n *for dimension* $n \geq 2$.
7. *The Brouwer fixed-point theorem* BFT_∞ *for the Hilbert cube* $[0,1]^{\mathbb{N}}$.
8. *Finding connectedness components of sets* $A \subseteq [0,1]^n$ *for* $n \geq 1$.
9. *The parallelization* $\widehat{\mathsf{IVT}}$ *of the intermediate value theorem.*
10. *Determinacy of Gale-Stewart games in* $2^{\mathbb{N}}$ *with closed winning sets.*

In the case of the Hahn-Banach theorem the underlying separable Banach space is part of the input information. No space of maximal complexity is known in this case. For certain spaces (such as computable Hilbert spaces) the Hahn-Banach theorem is computable. For two further theorems mentioned above we provide formalizations as problems.

1. $\mathsf{MAX} : \mathscr{C}[0,1] \rightrightarrows \mathbb{R}, f \mapsto \{x \in [0,1] : f(x) = \max f([0,1])\}$.
2. $\mathsf{BFT}_n : \mathscr{C}([0,1]^n, [0,1]^n) \rightrightarrows [0,1]^n, f \mapsto \{x \in [0,1]^n : f(x) = x\}$.

The Brouwer fixed-point theorem of dimension $n = 1$ is equivalent to the intermediate value theorem, i.e., $\mathsf{BFT}_1 \equiv_{\mathrm{sW}} \mathsf{IVT} \equiv_{\mathrm{sW}} \mathsf{CC}_{[0,1]}$. We note that classifications such as the one in Theorem 11.8.3 lead to simple proofs of classically known non-uniform results in computable analysis. We mention some examples.

Corollary 11.8.4 (Non-uniform results).

1. *There exists an infinite binary tree without computable paths (Kleene [70]).*
2. *There is a computable function* $f : [0,1] \to \mathbb{R}$ *that attains its maximum only at non-computable points* $x \in [0,1]$ *(Lacombe [77, Theorems VI and VII] and Specker [109]).*
3. *There is a computable function* $f : [0,1]^2 \to [0,1]^2$ *that has no computable fixed-point* $x \in [0,1]^2$ *(Orevkov [88] and Baigger [2]).*
4. *There is a computable sequence* $(f_n)_{n\in\mathbb{N}}$ *of functions* $f_n : [0,1] \to \mathbb{R}$ *with* $f_n(0) \cdot f_n(1) < 0$ *for all* $n \in \mathbb{N}$ *such that there is no computable sequence* $(x_n)_{n\in\mathbb{N}}$ *with* $f_n(x_n) = 0$ *(Pour-El and Richards [101, Example 8a]).*

Once one has one of these negative results, all the others follow immediately by Theorem 11.8.3. On the other hand, also positive non-uniform results can be derived from Theorem 11.8.3. For instance, every computable function $f : [0,1]^2 \to [0,1]^2$ has a low fixed-point. Such non-uniform results hold analogously for other classifications presented here, but we are not going to discuss them in detail.

A theorem that is often proved with the help of the Brouwer fixed-point theorem is the Nash equilibria existence theorem. Its computational content is significantly weaker than that of the Brouwer fixed-point theorem.

Theorem 11.8.5 (All-or-unique choice). *The following are strongly Weihrauch equivalent to each other:*

1. *The finite parallelization* $\mathsf{AoUC}^*_{[0,1]}$ *of all-or-unique choice.*
2. *The Nash equilibria existence theorem* NASH.

Here $\mathsf{NASH}_{n,m} : \mathbb{R}^{m\times n} \times \mathbb{R}^{m\times n} \rightrightarrows \mathbb{R}^n \times \mathbb{R}^m$ is the map that maps a bi-matrix game (A,B) to a pair of strategies that form a Nash equilibrium of (A,B), and $\mathsf{NASH} := \bigsqcup_{n,m\in\mathbb{N}} \mathsf{NASH}_{n,m}$.

The equivalence class of choice on Euclidean space contains a theorem that we mention in the following result without further definitions.

Theorem 11.8.6 (Choice on Euclidean space). *The following are Weihrauch equivalent to each other:*

1. *Choice on Euclidean space* $\mathsf{C}_{\mathbb{R}}$.
2. *Frostman's lemma on the existence of measures.*

The Vitali covering theorem is a theorem that has even been studied in three different logical versions. We consider $\mathsf{Int} := (\mathbb{Q}^2)^{\mathbb{N}}$ as the set of sequences $\mathscr{I} = (I_n)_{n\in\mathbb{N}}$ of rational intervals $I_n = (a,b)$. We say that $\mathscr{I}$ is a *Vitali cover* of a set $A \subseteq \mathbb{R}$ if for every $x \in A$ and $\varepsilon > 0$ there is some $n \in \mathbb{N}$ with $x \in I_n$ and $\operatorname{diam}(I_n) < \varepsilon$. We write $\mathscr{J} \sqsubseteq \mathscr{I}$ if $\mathscr{J}$ is a subsequence of $\mathscr{I}$ of pairwise disjoint intervals. We consider the following three formalizations of the Vitali covering theorem:

1. $\mathsf{VCT}_0 :\subseteq \mathsf{Int} \rightrightarrows \mathsf{Int}, \mathscr{I} \mapsto \{\mathscr{J} : \mathscr{J} \sqsubseteq \mathscr{I} \text{ with } \mu([0,1] \setminus \bigcup \mathscr{J}) = 0\}$ and $\operatorname{dom}(\mathsf{VCT}_0)$ contains all $\mathscr{I} \in \mathsf{Int}$ that are Vitali covers of $[0,1]$.
2. $\mathsf{VCT}_1 :\subseteq \mathsf{Int} \rightrightarrows [0,1], \mathscr{I} \mapsto [0,1] \setminus \bigcup \mathscr{I}$ and $\operatorname{dom}(\mathsf{VCT}_1)$ contains all $\mathscr{I} \in \mathsf{Int}$ that are Vitali covers of $\bigcup \mathscr{I}$ and without a $\mathscr{J} \sqsubseteq \mathscr{I}$ with $\mu([0,1] \setminus \bigcup \mathscr{J}) = 0$.
3. $\mathsf{VCT}_2 :\subseteq \mathsf{Int} \rightrightarrows [0,1], \mathscr{I} \mapsto \{x \in [0,1] : (\exists \varepsilon > 0)(\forall n)(x \notin I_n \text{ or } \operatorname{diam}(I_n) \geq \varepsilon)\}$ and $\operatorname{dom}(\mathsf{VCT}_2)$ contains all $\mathscr{I}$ without a $\mathscr{J} \sqsubseteq \mathscr{I}$ with $\mu([0,1] \setminus \bigcup \mathscr{J}) = 0$.

It turns out that VCT_0 is computable and VCT_1 and VCT_2 are equivalent to different versions of positive choice.

Theorem 11.8.7 (Positive choice). *The following are all strongly Weihrauch equivalent to each other:*

1. *Positive choice on the Cantor space* $\mathsf{PC}_{2^{\mathbb{N}}}$.
2. *Weak weak Kőnig's Lemma* WWKL.
3. *The Vitali covering theorem* VCT_1.

The following are strongly Weihrauch equivalent to each other:

1. *Positive choice on Euclidean space* $\mathsf{PC}_{\mathbb{R}}$.
2. *The Vitali covering theorem* VCT_2.

Convex choice is equivalent to the Browder-Göhde-Kirk fixed-point theorem, which is formalized as $\mathsf{BGK}_K :\subseteq \mathscr{C}(K,K) \rightrightarrows K, f \mapsto \{x \in K : f(x) = x\}$, where $K \subseteq H$ is compact and convex, H is a computable Hilbert space and $\operatorname{dom}(\mathsf{BGK}_K)$ consists of all non-expansive continuous maps $f : K \to K$. More general versions of the theorem have been studied, but for simplicity we state only this basic result.

Theorem 11.8.8 (Convex choice). *Let H be a computable Hilbert space and $K \subseteq H$ convex and computably compact. The following are Weihrauch equivalent to each other:*

1. *Convex choice* XC_K.
2. *The Browder-Göhde-Kirk fixed-point theorem* BGK_K *on* K.

Another important equivalence class is that of the limit map.

Theorem 11.8.9 (The limit). *The following are Weihrauch equivalent to each other:*

1. *The limit map* $\lim$ *on the Baire space (or every other rich computable metric space).*
2. *The monotone convergence theorem* $\mathsf{MCT} :\subseteq \mathbb{R}^{\mathbb{N}} \to \mathbb{R}, (x_n)_{n\in\mathbb{N}} \mapsto \sup_{n\in\mathbb{N}} x_n$.
3. *The operator of differentiation* $d :\subseteq \mathscr{C}[0,1] \to \mathscr{C}[0,1], f \mapsto f'$.
4. *The Fréchet-Riesz representation theorem for* ℓ_2.
5. *The Radon-Nikodym theorem.*
6. *The parallelization* $\widehat{\mathsf{BIM}}$ *of Banach's inverse mapping theorem.*
7. *Finding a basis of a countable vector space.*
8. *Finding a connected component of a countable graph.*
9. *The partial identity from infinitely differentiable functions to Schwartz functions.*

Of course, the Banach inverse mapping theorem can be replaced by any other problem from Theorem 11.8.2. We mention that the reduction $\lim \leq_{\mathrm{W}} d$ follows easily with Theorem 11.6.21. Several theorems also fall into the equivalence class of the jump of choice on the Cantor space. Here KL is defined as WKL but for finitely branching trees $T \subseteq \mathbb{N}^*$.

Theorem 11.8.10 (Jump of choice on the Cantor space). *The following are all strongly Weihrauch equivalent to each other:*

1. *The jump* $\mathsf{C}'_{2^{\mathbb{N}}}$ *of choice on the Cantor space.*
2. *Kőnig's lemma* KL.
3. *The Bolzano-Weierstraß theorem* $\mathsf{BWT}_{\mathbb{R}}$ *on Euclidean space.*
4. *The Arzelá-Ascoli theorem for functions* $f : [0,1] \to [0,1]$.
5. *Determinacy of Gale-Stewart games in* $2^{\mathbb{N}}$ *with winning sets that are differences of open sets.*

A natural problem that is known to be equivalent to higher jumps of choice on the Cantor space is the parallelization of Ramsey's theorem. We summarize some results on this theorem. $\mathsf{RT}^k_n : k^{[\mathbb{N}]^n} \rightrightarrows 2^{\mathbb{N}}$ denotes the problem that maps every coloring $c : [\mathbb{N}]^n \to k$ (of the n–element subsets of $\mathbb{N}$ with k colors) to an infinite set $H \subseteq \mathbb{N}$ that is homogeneous for c.

Theorem 11.8.11 (Ramsey's theorem). $\mathsf{C}^{(n)}_2 <_{\mathrm{W}} \mathsf{RT}^n_k <_{\mathrm{W}} \mathsf{RT}^n_{k+1} <_{\mathrm{W}} \mathsf{C}^{(n)}_{2^{\mathbb{N}}}$ *for all* $n,k \geq 2$. *The reductions also hold in the case* $n = 1$, *but the first one is not strict in this case.*

This result can be proved with the help of the squashing theorem (Theorem 11.4.6). Since $\mathsf{C}^{(n)}_{2^{\mathbb{N}}}$ is the parallelization of $\mathsf{C}^{(n)}_2$ we obtain the following corollary.

Corollary 11.8.12 (Ramsey's theorem). $\mathsf{C}_{2^{\mathbb{N}}}^{(n)} \equiv_{\mathrm{W}} \widehat{\mathsf{RT}_k^n}$ *for all* $n \geq 1$ *and* $k \geq 2$.

Higher levels of the Weihrauch lattices are not yet all too well explored. This is currently a topic of further research, and we mention one result along these lines.

Theorem 11.8.13 (Choice on the Baire space). *The following are Weihrauch equivalent to each other:*

1. *Choice on the Baire space* $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$.
2. *The perfect subtree theorem.*

At the end of this section we demonstrate how some problems from computability theory can be classified in the Weihrauch lattice. We consider in particular the following (for $X \subseteq \mathbb{N}$ with at least two elements and a standard numbering φ^p of the computable functions on natural numbers relative to p):

1. $\mathsf{DNC}_X : \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}, p \mapsto \{q \in X^{\mathbb{N}} : (\forall n)\ \varphi_n^p(n) \neq q(n)\}$.
2. $1\text{-}\mathsf{GEN} : 2^{\mathbb{N}} \rightrightarrows 2^{\mathbb{N}}, p \mapsto \{q : q \text{ is 1–generic relative to } p\}$.
3. $\mathsf{MLR} : 2^{\mathbb{N}} \rightrightarrows 2^{\mathbb{N}}, p \mapsto \{q : q \text{ is Martin-Löf random relative to } p\}$.
4. $\mathsf{PA} : 2^{\mathbb{N}} \rightrightarrows 2^{\mathbb{N}}, p \mapsto \{q : q \text{ is of PA degree relative to } p\}$.
5. $\mathsf{COH} : (2^{\mathbb{N}})^{\mathbb{N}} \rightrightarrows 2^{\mathbb{N}}, (R_i)_{i\in\mathbb{N}} \mapsto \{A : A \text{ is cohesive for } (R_i)_{i\in\mathbb{N}}\}$.

The first observation is that DNC_n is just the parallelization of ACC_n.

Theorem 11.8.14 (Diagonal non-computability). $\mathsf{DNC}_n \equiv_{\mathrm{sW}} \widehat{\mathsf{ACC}_n}$ *for all* $n \geq 2$ *and* $n = \mathbb{N}$.

The jump BCT_0' of the computable version of the Baire category theorem BCT_0 is closely related to 1–genericity and the problem $\Pi_1^0\mathsf{G}$ of Π_1^0–genericity, which we do not define here.

Theorem 11.8.15 (Genericity). $1\text{-}\mathsf{GEN} <_{\mathrm{W}} \mathsf{BCT}_0' \equiv_{\mathrm{sW}} \Pi_1^0\mathsf{G} <_{\mathrm{W}} \mathsf{L}$.

We note that all the problems from computability theory mentioned here, except DNC_n, are densely realized. Hence we can apply Proposition 11.4.13.

Proposition 11.8.16. $1\text{-}\mathsf{GEN}$, MLR, PA, COH *and* BCT_0' *are densely realized and hence* $\mathsf{ACC}_{\mathbb{N}}$ *and* C_2 *are not Weihrauch reducible to any of them.*

This means that these problems are very different from all the theorems from analysis mentioned above that are all above C_2 in the Weihrauch lattice. Hence, it is interesting that some of these densely realized problems can be characterized as implications (i.e., as "quotients") of problems above C_2.

Theorem 11.8.17 (Randomness, Peano arithmetic, cohesiveness). *We obtain:*

1. $\mathsf{MLR} \equiv_{\mathrm{W}} (\mathsf{C}_{\mathbb{N}} \to \mathsf{WWKL}) \equiv_{\mathrm{W}} (\mathsf{C}_{\mathbb{N}} \to \mathsf{PC}_{2^{\mathbb{N}}})$.
2. $\mathsf{PA} \equiv_{\mathrm{W}} (\mathsf{C}_{\mathbb{N}}' \to \mathsf{WKL}) \equiv_{\mathrm{W}} (\mathsf{C}_{\mathbb{N}}' \to \mathsf{C}_{2^{\mathbb{N}}})$.
3. $\mathsf{COH} \equiv_{\mathrm{W}} (\lim \to \mathsf{KL}) \equiv_{\mathrm{W}} (\lim \to \mathsf{C}_{2^{\mathbb{N}}}')$.

We close this section by mentioning that one can apply results from computability theory such as the theorem of van Lambalgen to conclude that some of the above-mentioned problems are closed under composition.

Proposition 11.8.18. MLR *and* $1\text{-}\mathsf{GEN}$ *are closed under compositional product* $*$.

Bibliographic Remarks

The intermediate value theorem and theorems from functional analysis were studied by Brattka and Gherardi [17, 12, 10]. The Lebesgue covering lemma for $[0,1]$ has been classified by Brattka, Gherardi and Hölzl [19]. The equivalence of the Hahn-Banach theorem and weak Kőnig's lemma was proved by Gherardi and Marcone [46, 11]. The Brouwer fixed-point theorem and the problem of finding connectedness components was studied by Brattka, Le Roux, Miller and Pauly [27]. The theorem of the maximum has been studied by Brattka [13] and the Fréchet-Riesz representation theorem by Brattka and Yoshikawa [13, 33]. The classification of the Nash equilibria existence theorem is due to Pauly [91, 93]. Frostman's lemma was studied by Pauly and Fouché [99] and Vitali's covering theorem by Brattka, Gherardi, Hölzl and Pauly [20]. The Browder-Göhde-Kirk fixed-point theorem was classified by Neumann [85]. The operator of differentiation was studied by von Stein [110], and the degrees of many operations on sets that are not mentioned here were classified by Brattka and Gherardi [16]. The identities to analytic and Schwartz functions have been classified by Pauly and Steinberg [100]. The analysis of the Radon-Nikodym theorem is due to Hoyrup, Rojas and Weihrauch [60]. The problems of finding a basis of a countable vector space and of finding a connected component of a countable graph were studied by Gura, Hirst and Mummert [47, 59]. The Bolzano-Weierstraß theorem was studied by Brattka, Gherardi and Marcone [21]. Kőnig's lemma was studied by Brattka and Rakotoniaina [31] and Gale-Stewart games by Le Roux and Pauly [80]. Ramsey's theorem was studied in the Weihrauch lattice by Dorais, Dzhafarov, Hirst, Mileti and Shafer [38, 40, 41], by Brattka and Rakotoniaina [31, 102], by Patey [89] and by Hirschfeldt and Jockusch [58]. The uniform content of problems for partial and linear orders that are closely related to Ramsey's theorem for pairs was studied by Astor, Dzhafarov, Solomon and Suggs [1]. The classification of the prefect subtree theorem was initiated by Marcone [26], and the result mentioned here is unpublished. The results on problems from computability theory including a systematic study of the Baire category theorem are due to Brattka, Hendtlass and Kreuzer [22, 23] and Brattka and Pauly [29]. The problem of diagonally non-computable functions was also studied by Higuchi and Kihara [54].

11.9 Relations to Other Theories

In this section we discuss very briefly the relation between Weihrauch complexity and other theories and we provide some further references.

11.9.1 Linear Logic

There is an apparent similarity between some algebraic operations on problems and the resource-oriented interpretation of some logical operations in (intuitionistic) linear logic that was noticed early on. Table 11.1 provides a dictionary on these relations.

However, it seems that other algebraic operations on problems, such as the compositional product $*$, do not have any obvious counterpart in the standard approach to linear logic. The compositional product could be seen as a non-commutative conjunction. There does not seem to be any straightforward interpretation of the Weihrauch lattice as a model for (intuitionistic) linear logic.

logical operation in linear logic	algebraic operation on problems
$\otimes$ multiplicative conjunction	$\times$ product
& additive conjunction	$\sqcup$ coproduct
$\oplus$ additive disjunction	$\sqcap$ infimum
$\parr$ multiplicative disjunction	$+$ sum
! bang	$\widehat{\ }$parallelization, * finite parallelization

Table 11.1 Linear logic versus the algebra of problems.

Several researchers have independently noticed that Gödel's Dialectica interpretation has some formal similarity to Weihrauch reducibility. This observation has not yet been formally exploited.

11.9.2 Medvedev Lattice, Many-One and Turing Semilattices

The Medvedev lattice has also been considered as a calculus of problems. Here problems are understood to be subsets $A, B \subseteq \mathbb{N}^{\mathbb{N}}$ of the Baire space and A is called *Medvedev reducible* to B, in symbols $A \leq_{\mathrm{s}} B$, if there exists a partial computable function $F :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ with $B \subseteq \mathrm{dom}(F)$ such that $F(B) \subseteq A$. The supremum operation of this lattice is defined by $A \oplus B := \langle A, B\rangle$ and the infimum operation by $A \otimes B := 0A \cup 1B$.

The relation between the Weihrauch lattice and the Medvedev lattice can be expressed from both perspectives:

1. The Medvedev lattice is a special case of the Weihrauch lattice for problems $f :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}$ that are constant.
2. The Weihrauch lattice is a generalization of the Medvedev lattice for "relativized" problems $A_p \subseteq \mathbb{N}^{\mathbb{N}}$ that depend on a parameter $p \in \mathbb{N}^{\mathbb{N}}$.

This point of view translates into a formal embedding of the Medvedev preorder into the Weihrauch preorder (the first result mentioned in Theorem 11.9.1). In fact, we can embed the Medvedev lattice also with order reversed into the Weihrauch lattice, and we list both embeddings here.

Theorem 11.9.1 (Embedding the Medvedev lattice). *Let $A, B \subseteq \mathbb{N}^{\mathbb{N}}$.*

1. *$c_A : \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}, p \mapsto A$ satisfies $A \leq_{\mathrm{s}} B \iff c_A \leq_{\mathrm{W}} c_B$ with $c_{A\oplus B} \equiv_{\mathrm{W}} c_A \times c_B$ and $c_{A\otimes B} \equiv_{\mathrm{W}} c_A \sqcap c_B$.*
2. *$d_A :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}, p \mapsto \widehat{0}$ with $\mathrm{dom}(d_A) := A$ satisfies $A \leq_{\mathrm{s}} B \iff d_B \leq_{\mathrm{W}} d_A$ with $d_{A\oplus B} \equiv_{\mathrm{W}} d_A \sqcap d_B$ and $d_{A\otimes B} \equiv_{\mathrm{W}} d_A \sqcup d_B$.*

In both cases all Weihrauch reductions and equivalences can be replaced by strong ones, in which case $\sqcup$ has to be replaced by $\boxplus$ in 2.

The second, reverse embedding is even a lattice embedding since it preserves suprema and infima in the reverse order. The first embedding is also a lattice embedding if considered as an embedding into the parallelized Weihrauch degrees. These

embeddings were studied by Brattka and Gherardi [18], Higuchi and Pauly [55] and Dzhafarov [42]. Since the Turing degrees and the enumeration degrees can be embedded into the Medvedev lattice, it follows that they can also be embedded into the Weihrauch lattice via the abovementioned embeddings.

We note that the Medvedev lattice has been used by Downey, Greenberg, Jockusch, Milans, Lewis and others [39, 63] in order to study problems from computability theory, such as MLR, 1-GEN, PA and DNC_n in their unrelativized form (for computable inputs). The advantage of the Weihrauch lattice is that these problems can be studied in this lattice together with problems such as WKL and WWKL that depend on parameters (i.e., the input tree) in an essential way. Finally, we mention that the Muchnik lattice, which is the non-uniform counterpart of the Medvedev lattice, has also been used to classify problems, see Simpson [107] .

Also the many-one semilattice can be embedded into the Weihrauch lattice, albeit in a slightly less natural way than the Turing semilattice. The construction starts with a non-canonical choice of two Turing incomparable points. As usual we denote *many-one reducibility* between sets $A, B \subseteq \mathbb{N}$ by $\leq_{\mathrm{m}}$ and we recall that $A \oplus B := \{2n : n \in A\} \cup \{2n+1 : n \in B\}$ is the supremum with respect to many-one reducibility.

Proposition 11.9.2 (Embedding of the many-one semilattice). *Let $p, q \in \mathbb{N}^{\mathbb{N}}$ be Turing incomparable and, for $A \subseteq \mathbb{N}$, define $m_A : \mathbb{N} \to \{p, q\}$ by $m_A(n) = p :\iff n \in A$. Then we obtain $A \leq_{\mathrm{m}} B \iff m_A \leq_{\mathrm{W}} m_B$ and $m_{A \oplus B} \equiv_{\mathrm{W}} m_A \sqcup m_B$ for all $A, B \subseteq \mathbb{N}$, i.e., $A \mapsto m_A$ is a join-semilattice embedding.*

11.9.3 Reverse Mathematics

Reverse mathematics is a proof-theoretic approach that aims to classify theorems according to the axioms that are needed to prove these theorems in second-order arithmetic [106]. Many theorems from various areas of mathematics have been classified in this approach. Most axiom systems that are used in reverse mathematics have counterparts in the Weihrauch lattice (see also Figure 11.2):

- $\mathsf{B}\Sigma^0_n$ (Σ^0_n–boundedness): $\mathsf{B}\Sigma^0_2$ is equivalent to the *regularity principle* $\mathsf{R}\Sigma^0_1$ over a very weak system [48], and it corresponds to $\mathsf{K}'_{\mathbb{N}}$ by Theorem 11.7.52. Hence $\mathsf{B}\Sigma^0_n$ can be seen as the counterpart of $\mathsf{K}^{(n-1)}_{\mathbb{N}}$.
- $\mathsf{I}\Sigma^0_n$ (Σ^0_n–induction) is equivalent to the *least number principle* $\mathsf{L}\Pi^0_n$ over a very weak system [48]. $\mathsf{L}\Pi^0_1$ directly translates into $\min^{\mathrm{c}}$ as a problem and hence $\mathsf{I}\Sigma^0_n$ corresponds to $\mathsf{C}^{(n-1)}_{\mathbb{N}}$ by Theorem 11.8.2.
- RCA^*_0 (recursive comprehension) stands for the usual system RCA_0 but with $\mathsf{I}\Sigma^0_0$ instead of $\mathsf{I}\Sigma^0_1$. It corresponds to C_1 (the computable problems).
- WKL^*_0 and WWKL^*_0, by which we mean RCA^*_0 plus weak Kőnig's lemma and weak weak Kőnig's lemma, respectively, correspond directly to the problems WKL and WWKL.

- ACA_0 (arithmetic comprehension) corresponds to the problems lim (and its finite compositions $\lim^{[n]}$ with $n \in \mathbb{N}$). Sometimes, a uniform version ACA_0' of ACA_0 is used [57], which corresponds to $\bigsqcup_{n\in\mathbb{N}} \lim^{[n]}$.
- ATR_0 (arithmetical transfinite recursion) corresponds to $\mathsf{UC}_{\mathbb{N}^{\mathbb{N}}}$ and $\mathsf{C}_{\mathbb{N}^{\mathbb{N}}}$ (see Theorem 11.8.13). This topic is still very much research in progress.

Counterparts in the Weihrauch lattice of higher systems such as Π^1_1-CA_0 (Π^1_1–comprehension) have not yet been systematically studied. By Theorem 11.7.55 we have $\mathsf{K}^{(n)}_{\mathbb{N}} <_{\mathrm{W}} \mathsf{C}^{(n)}_{\mathbb{N}} <_{\mathrm{W}} \mathsf{K}^{(n+1)}_{\mathbb{N}}$ in analogy to $\mathsf{B}\Sigma^0_n \leftarrow \mathsf{I}\Sigma^0_n \leftarrow \mathsf{B}\Sigma^0_{n+1}$.

Reverse mathematics is based on a proof-theoretic approach, whereas classifications in the Weihrauch lattice are based on a computational approach. Besides this we note the following distinguishing features:

1. *Resource sensitivity*: classifications in reverse mathematics do not distinguish between a single application, a finite number of consecutive applications or a finite number of parallel applications of a theorem, since classical logic is used (as opposed to linear logic).
2. *Uniformity*: classifications in reverse mathematics only capture the non-uniform content of problems, i.e., the way output parameters depend on input parameters in the worst case. Again this is due to the usage of classical logic (as opposed to intuitionistic logic).

For instance, a number of theorems that are non-uniformly computable in the sense that there is a computable output for every computable input are provable over RCA_0 in reverse mathematics, even though they are not computable in a uniform way. This includes the intermediate value theorem IVT, the Baire category theorem BCT_1 and others. Due to the lack of uniformity reverse mathematics also cannot distinguish between theorems and their contrapositive forms. For instance the version HBC_0 of the Heine-Borel covering theorem is computable, while $\mathsf{HBC}_1 \equiv_{\mathrm{W}} \mathsf{WKL}$. In reverse mathematics, the Heine-Borel theorem is equivalent to WKL_0 over RCA_0 irrespectively of whether we consider the analogue of HBC_0 or HBC_1. In other words: classifications in reverse mathematics automatically capture the most complicated contrapositive form.

It is remarkable that despite these explicable differences most classifications in the Weihrauch lattice can be seen as uniform and resource sensitive refinements of classifications in reverse mathematics. This seems to confirm a "computations as proofs" paradigm (as opposed to the well-known "proofs as computations" paradigm in intuitionistic logic).

11.9.4 Constructive Reverse Mathematics

Constructive reverse mathematics, as proposed by Ishihara [61], classifies problems in the Bishop approach to constructive analysis, which is based on intuitionistic logic. Due to the usage of intuitionistic logic this approach is fully uniform, but it is

even less resource sensitive compared to classical reverse mathematics. This is due to the fact that typically the *axiom of countable choice* can be used freely, which amounts to a free usage of parallelization in the Weihrauch lattice. In this sense, the Weihrauch complexity approach is closer to a hypothetical version of constructive reverse mathematics with intuitionistic linear logic. The classifications in constructive reverse mathematics are captured by their equivalence to certain constructively unacceptable principles:

1. LLPO (the *lesser limited principle of omniscience*) is the theorem that corresponds to our problem LLPO. In the presence of countable choice it corresponds to WKL by Theorem 11.7.23.
2. LPO (the *limited principle of omniscience*) is the theorem that corresponds to our problem LPO. In the presence of countable choice and due to the availability of composition it corresponds to $\lim^{(n)}$ with $n \in \mathbb{N}$ by Theorem 11.6.7 (and hence to ACA_0 in classical reverse mathematics).
3. MP (*Markov's principle*), BD-$\mathbb{N}$ (the *boundedness problem*) and some other principles that are rejected in constructive analysis correspond to computable (and hence continuous) problems in the Weihrauch lattice.

In conclusion, this means that the Weihrauch complexity approach is finer than constructive reverse mathematics in terms of resource sensitivity, but coarser when it comes to distinctions that are based on computable principles such as MP and BD-$\mathbb{N}$. In order to translate these heuristic observations into formal theorems, one needs to fix an axiomatic framework for constructive analysis. Some results in this direction have been obtained by Kuyper [75], Fujiwara [43] and Uftring [112].

11.9.5 Other Reducibilities

Hirschfeldt and Jockusch [57, 58, 108] have introduced a number of further reducibilities that are related to Weihrauch reducibility. For one, there are non-uniform versions of Weihrauch reducibility and strong Weihrauch reducibility, which are called *computable reducibility* and *strong computable reducibility*, in symbols $\leq_{\mathrm{c}}$ and $\leq_{\mathrm{sc}}$, as well as a reducibility $\leq_{\omega}$ that is based on Turing ideals. On the other hand, they introduced a concept of *generalized Weihrauch reducibility* that has a built-in closure under composition. This operation can be formalized as a closure operator $f \mapsto f^{\diamond}$ in the Weihrauch lattice. Likewise, Brattka and Gherardi [18] and Higuchi and Pauly [55] studied variants of Weihrauch reducibility with a built-in parallelization. These and further reducibilities allow one to interpolate between Weihrauch complexity and reverse mathematics in the sense that one can choose a reduction that captures a particular degree of uniformity and resource sensitivity (see Figure 11.3).

Yet another related reducibility that originates from descriptive set theory is *Wadge reducibility*, which is defined via preimages. Given $A, B \subseteq \mathbb{N}^{\mathbb{N}}$, we say that A is *Wadge reducible* to B, in symbols $A \leq_{\mathrm{W}} B$, if there exists a continuous function

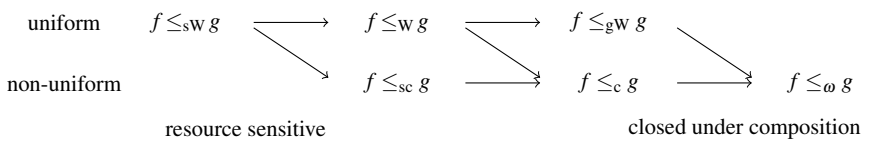


Fig. 11.3 Implications between notions of reducibility.

$f : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ such that $A = f^{-1}(B)$. Hence, Wadge reducibility is the (topological) analogue of many-one reducibility on the Baire space. Weihrauch reducibility can be seen as a (computable) analogue of this reduction for multi-valued functions.

In early work by Weihrauch [114, 115] and by Hertling [50] mostly the continuous version of Weihrauch reducibility was considered. In particular, Hertling [50, 51] completely characterized continuous (strong) Weihrauch and Wadge degrees of certain functions with discrete image in terms of preorders on labeled forests. Kudinov, Selivanov and Zhukov [74] and Hertling and Selivanov [52] have studied the decidability and complexity of some initial segments of these preorders. Such preorders and versions of Weihrauch reducibility have also been used in descriptive set theory, e.g., by Carroy [37].

11.9.6 Descriptive Set Theory

Descriptive set theory studies the complexity of subsets of and functions between separable complete metric spaces. Wadge reducibility has been established as a critical tool here, and reasonable classes of subsets are typically closed downwards under Wadge reducibility. For functions, Weihrauch reducibility can play the analogous role. As demanded by Moschovakis [81], this treatment covers both the effective and the non-effective case simultaneously, with the former implying the latter via relativization.

Many typical classes of functions even have complete problems under Weihrauch reducibility. Theorem 11.6.5 provides an example; another one is related to $(\Sigma^0_{n+2}, \Sigma^0_{n+2})$–measurability, which is defined such that preimages of Σ^0_{n+2}–sets are Σ^0_{n+2}–sets (see Pauly and de Brecht [97] and Kihara [67]).

Theorem 11.9.3 (Effective $(\Sigma^0_{n+2}, \Sigma^0_{n+2})$–measurability). *$f \leq_{\mathrm{W}} \mathsf{C}^{(n)}_{\mathbb{N}} \iff$ f is effectively $(\Sigma^0_{n+2}, \Sigma^0_{n+2})$–measurable, for all $f : X \to Y$ on complete computable metric spaces X, Y with $n = 0$ and for $f : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ with $n \in \mathbb{N}$.*

There is a subtle but crucial issue with relativization here: relativizing the theorem covers the case where the preimage map of f from Σ^0_{n+2}–sets to Σ^0_{n+2}–sets is continuous, rather than merely being well defined. For $n \in \{0, 1\}$ the theorem of Jayne and Rogers [62] and a theorem by Semmes [105] show that these cases are equivalent. For $n > 1$, the question of whether the cases are equivalent is open and

equivalent to the generalized conjecture of Jayne and Rogers. This is discussed in some more detail in [94].

Weihrauch complete problems for function classes correspond to game characterizations in descriptive set theory. A general account of the latter is provided by Motto Ros [82], and the link to Weihrauch reducibility is made by Nobrega and Pauly [87].

More generally, the theory of Weihrauch degrees is closely linked to a programme to extend descriptive set theory from Polish spaces to larger classes of spaces, such as represented spaces. Such an endeavor was called for and started by Selivanov [104]. De Brecht introduced the quasi-Polish spaces [34] and demonstrated that many results from descriptive set theory remain valid in this setting. A further extension is possible using the formalism of jump operators (de Brecht [35]) or computable endofunctors (Pauly and de Brecht [98]), both of which are closely related to each other and to Weihrauch degrees.

11.9.7 Other Models of Computability

We have already seen in Sections 11.6 and 11.7 that other models of computability can be characterized in the Weihrauch lattice. This includes the classes of problems that are computable with finitely many mind changes, limit computable, nondeterministically computable and Las Vegas computable.

There are completely different algebraic models of computability such as the Blum-Shub-Smale machines [4] (BSS machines). Due to their algebraic nature, the class of functions computable by these machines lack certain completeness properties and cannot be characterized exactly in the Weihrauch lattice. However, some tight upper bounds have been found by Neumann and Pauly [86].

Theorem 11.9.4 (Algebraic computation). *If $f :\subseteq \mathbb{R}^* \to \mathbb{R}^*$ is computable on a BSS machine, then $f \leq_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$ and there is a function $f :\subseteq \mathbb{R}^* \to \mathbb{R}^*$ that is computable on a BSS machine and satisfies $f \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$.*

Hertling and Weihrauch [53, 49] have studied how the number of tests that are performed is related to degeneracies in computations. Yet a further class of machines can be obtained if one allows infinite computation time of higher order. Weihrauch computability was generalized to this context by Carl [36] and Galeotti and Nobrega [44].

Acknowledgements Vasco Brattka has been supported by the National Research Foundation of South Africa (Grant Number 115269). Guido Gherardi has been supported by MIUR-PRIN project *Analysis of Program Analyses* (ASPRA, ID 201784YSZ5_004).

References

1. Astor, E.P., Dzhafarov, D.D., Solomon, R., Suggs, J.: The uniform content of partial and linear orders. Annals of Pure and Applied Logic **168**(6), 1153 – 1171 (2017)
2. Baigger, G.: Die Nichtkonstruktivität des Brouwerschen Fixpunktsatzes. Archive for Mathematical Logic **25**, 183–188 (1985)
3. Bienvenu, L., Kuyper, R.: Parallel and serial jumps of Weak Weak König's Lemma. In: A. Day, M. Fellows, N. Greenberg, B. Khoussainov, A. Melnikov, F. Rosamond (eds.) Computability and Complexity: Essays Dedicated to Rodney G. Downey on the Occasion of His 60th Birthday, *Lecture Notes in Computer Science*, vol. 10010, pp. 201–217. Springer, Cham (2017)
4. Blum, L., Cucker, F., Shub, M., Smale, S.: Complexity and Real Computation. Springer, New York (1998)
5. Brattka, V.: Grade der Nichtstetigkeit in der Analysis. Diplomarbeit. Fachbereich Informatik, FernUniversität Hagen (1993)
6. Brattka, V.: Recursive and computable operations over topological structures. Ph.D. thesis. Department of Computer Science, University of Hagen, Hagen, Germany (1998)
7. Brattka, V.: Computable invariance. Theoretical Computer Science **210**, 3–20 (1999)
8. Brattka, V.: Computability over topological structures. In: S.B. Cooper, S.S. Goncharov (eds.) Computability and Models, pp. 93–136. Kluwer Academic Publishers, New York (2003)
9. Brattka, V.: Effective Borel measurability and reducibility of functions. Mathematical Logic Quarterly **51**(1), 19–44 (2005)
10. Brattka, V.: Computable versions of the uniform boundedness theorem. In: Z. Chatzidakis, P. Koepke, W. Pohlers (eds.) Logic Colloquium 2002, *Lecture Notes in Logic*, vol. 27, pp. 130–151. Association for Symbolic Logic, Urbana (2006)
11. Brattka, V.: Borel complexity and computability of the Hahn-Banach Theorem. Archive for Mathematical Logic **46**(7–8), 547–564 (2008)
12. Brattka, V.: A computable version of Banach's inverse mapping theorem. Annals of Pure and Applied Logic **157**, 85–96 (2009)
13. Brattka, V.: Computability and analysis, a historical approach. In: A. Beckmann, L. Bienvenu, N. Jonoska (eds.) Pursuit of the Universal, *Lecture Notes in Computer Science*, vol. 9709, pp. 45–57. Springer, Switzerland (2016), 12th Conference on Computability in Europe, CiE 2016, Paris, France, June 27 - July 1, 2016
14. Brattka, V., de Brecht, M., Pauly, A.: Closed choice and a uniform low basis theorem. Annals of Pure and Applied Logic **163**, 986–1008 (2012)
15. Brattka, V., Cettolo, A., Gherardi, G., Marcone, A., Schröder, M.: Addendum to: "The Bolzano-Weierstrass theorem is the jump of weak Kőnig's lemma". Annals of Pure and Applied Logic **168**(8), 1605–1608 (2017)
16. Brattka, V., Gherardi, G.: Borel complexity of topological operations on computable metric spaces. Journal of Logic and Computation **19**(1), 45–76 (2009)
17. Brattka, V., Gherardi, G.: Effective choice and boundedness principles in computable analysis. The Bulletin of Symbolic Logic **17**(1), 73–117 (2011)
18. Brattka, V., Gherardi, G.: Weihrauch degrees, omniscience principles and weak computability. The Journal of Symbolic Logic **76**(1), 143–176 (2011)
19. Brattka, V., Gherardi, G., Hölzl, R.: Probabilistic computability and choice. Information and Computation **242**, 249–286 (2015)
20. Brattka, V., Gherardi, G., Hölzl, R., Pauly, A.: The Vitali covering theorem in the Weihrauch lattice. In: A. Day, M. Fellows, N. Greenberg, B. Khoussainov, A. Melnikov, F. Rosamond (eds.) Computability and Complexity: Essays Dedicated to Rodney G. Downey on the Occasion of His 60th Birthday, *Lecture Notes in Computer Science*, vol. 10010, pp. 188–200. Springer, Cham (2017)
21. Brattka, V., Gherardi, G., Marcone, A.: The Bolzano-Weierstrass theorem is the jump of weak Kőnig's lemma. Annals of Pure and Applied Logic **163**, 623–655 (2012)

22. Brattka, V., Hendtlass, M., Kreuzer, A.P.: On the uniform computational content of computability theory. Theory of Computing Systems **61**(4), 1376–1426 (2017)
23. Brattka, V., Hendtlass, M., Kreuzer, A.P.: On the uniform computational content of the Baire category theorem. Notre Dame Journal of Formal Logic **59**(4), 605–636 (2018)
24. Brattka, V., Hertling, P., Weihrauch, K.: A tutorial on computable analysis. In: S.B. Cooper, B. Löwe, A. Sorbi (eds.) New Computational Paradigms: Changing Conceptions of What Is Computable, pp. 425–491. Springer, New York (2008)
25. Brattka, V., Hölzl, R., Kuyper, R.: Monte Carlo computability. In: H. Vollmer, B. Vallée (eds.) 34th Symposium on Theoretical Aspects of Computer Science (STACS 2017), *Leibniz International Proceedings in Informatics (LIPIcs)*, vol. 66, pp. 17:1–17:14. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2017)
26. Brattka, V., Kawamura, A., Marcone, A., Pauly, A. (eds.): Measuring the Complexity of Computational Content: Weihrauch Reducibility and Reverse Analysis (Dagstuhl Seminar 15392), *Dagstuhl Reports*, vol. 5. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2016)
27. Brattka, V., Le Roux, S., Miller, J.S., Pauly, A.: Connected choice and the Brouwer fixed point theorem. Journal of Mathematical Logic **19**(1), 1–46 (2019)
28. Brattka, V., Pauly, A.: Computation with advice. In: X. Zheng, N. Zhong (eds.) CCA 2010, Proceedings of the Seventh International Conference on Computability and Complexity in Analysis, *Electronic Proceedings in Theoretical Computer Science*, vol. 24, pp. 41–55 (2010)
29. Brattka, V., Pauly, A.: On the algebraic structure of Weihrauch degrees. Logical Methods in Computer Science **14**(4:4), 1–36 (2018)
30. Brattka, V., Presser, G.: Computability on subsets of metric spaces. Theoretical Computer Science **305**, 43–76 (2003)
31. Brattka, V., Rakotoniaina, T.: On the uniform computational content of Ramsey's theorem. Journal of Symbolic Logic **82**(4), 1278–1316 (2017)
32. Brattka, V., Weihrauch, K.: Computability on subsets of Euclidean space I: Closed and compact subsets. Theoretical Computer Science **219**, 65–93 (1999)
33. Brattka, V., Yoshikawa, A.: Towards computability of elliptic boundary value problems in variational formulation. Journal of Complexity **22**(6), 858–880 (2006)
34. de Brecht, M.: Quasi-Polish spaces. Ann. Pure Appl. Logic **164**(3), 356–381 (2013)
35. de Brecht, M.: Levels of discontinuity, limit-computability, and jump operators. In: V. Brattka, H. Diener, D. Spreen (eds.) Logic, Computation, Hierarchies, *Ontos Mathematical Logic*, pp. 93–122. Walter de Gruyter, Boston (2014)
36. Carl, M.: Generalized effective reducibility. In: A. Beckmann, L. Bienvenu, N. Jonoska (eds.) Pursuit of the Universal, *Lecture Notes in Computer Science*, vol. 9709, pp. 225–233. Springer, Switzerland (2016), 12th Conference on Computability in Europe, CiE 2016, Paris, France, June 27 - July 1, 2016
37. Carroy, R.: A quasi-order on continuous functions. Journal of Symbolic Logic **78**(2), 663–648 (2013)
38. Dorais, F.G., Dzhafarov, D.D., Hirst, J.L., Mileti, J.R., Shafer, P.: On uniform relationships between combinatorial problems. Transactions of the American Mathematical Society **368**(2), 1321–1359 (2016)
39. Downey, R.G., Greenberg, N., Jockusch, C.G., Jr., Milans, K.G.: Binary subtrees with few labeled paths. Combinatorica **31**(3), 285–303 (2011)
40. Dzhafarov, D.D.: Cohesive avoidance and strong reductions. Proceedings of the American Mathematical Society **143**(2), 869–876 (2015)
41. Dzhafarov, D.D.: Strong reductions between combinatorial principles. Journal of Symbolic Logic **81**(4), 1405–1431 (2016)
42. Dzhafarov, D.D.: Joins in the strong Weihrauch degrees. Mathematical Research Letters **26**(3), 749–767 (2019)
43. Fujiwara, M.: Weihrauch and constructive reducibility between existence statements. Computability **10**(to appear) (2021)

44. Galeotti, L., Nobrega, H.: Towards computable analysis on the generalized real line. In: J. Kari, F. Manea, I. Petre (eds.) Unveiling Dynamics and Complexity, *Lecture Notes in Computer Science*, vol. 10307, pp. 246–257. Springer, Cham (2017), 13th Conference on Computability in Europe, CiE 2017, Turku, Finland, June 12-16, 2017
45. Gherardi, G.: Effective Borel degrees of some topological functions. Mathematical Logic Quarterly **52**(6), 625–642 (2006)
46. Gherardi, G., Marcone, A.: How incomputable is the separable Hahn-Banach theorem? Notre Dame Journal of Formal Logic **50**(4), 393–425 (2009)
47. Gura, K., Hirst, J.L., Mummert, C.: On the existence of a connected component of a graph. Computability **4**(2), 103–117 (2015)
48. Hájek, P., Pudlák, P.: Metamathematics of First-Order Arithmetic. *Perspectives in Mathematical Logic*. Springer-Verlag, Berlin (1993)
49. Hertling, P.: Topological complexity with continuous operations. Journal of Complexity **12**, 315–338 (1996)
50. Hertling, P.: Unstetigkeitsgrade von Funktionen in der effektiven Analysis. Dissertation. Fachbereich Informatik, FernUniversität Hagen (1996)
51. Hertling, P.: Forests describing Wadge degrees and topological Weihrauch degrees of certain classes of functions and relations. Computability **9**(3-4), 249–307 (2020)
52. Hertling, P., Selivanov, V.: Complexity issues for preorders on finite labeled forests. In: V. Brattka, H. Diener, D. Spreen (eds.) Logic, Computation, Hierarchies, *Ontos Mathematical Logic*, pp. 165–190. Walter de Gruyter, Boston (2014)
53. Hertling, P., Weihrauch, K.: Levels of degeneracy and exact lower complexity bounds for geometric algorithms. In: Proceedings of the Sixth Canadian Conference on Computational Geometry, pp. 237–242. University of Saskatchewan (1994), Saskatoon, Saskatchewan, August 2–6, 1994
54. Higuchi, K., Kihara, T.: Inside the Muchnik degrees II: The degree structures induced by the arithmetical hierarchy of countably continuous functions. Annals of Pure and Applied Logic **165**(6), 1201–1241 (2014)
55. Higuchi, K., Pauly, A.: The degree structure of Weihrauch reducibility. Log. Methods Comput. Sci. **9**(2), 2:02, 17 (2013)
56. Hirsch, M.D.: Applications of topology to lower bound estimates in computer science. Ph.D. thesis. University of California, Berkeley (1990)
57. Hirschfeldt, D.R.: Slicing the Truth: On the Computable and Reverse Mathematics of Combinatorial Principles, *Lecture Notes Series, Institute for Mathematical Sciences, National University of Singapore*, vol. 28. World Scientific, Singapore (2015)
58. Hirschfeldt, D.R., Jockusch, C.G.: On notions of computability-theoretic reduction between Π^1_2 principles. Journal of Mathematical Logic **16**(1), 1650,002, 59 (2016)
59. Hirst, J.L., Mummert, C.: Reverse mathematics of matroids. In: A. Day, M. Fellows, N. Greenberg, B. Khoussainov, A. Melnikov, F. Rosamond (eds.) Computability and Complexity: Essays Dedicated to Rodney G. Downey on the Occasion of His 60th Birthday, *Lecture Notes in Computer Science*, vol. 10010, pp. 143–159. Springer, Cham (2017)
60. Hoyrup, M., Rojas, C., Weihrauch, K.: Computability of the Radon-Nikodym derivative. Computability **1**(1), 3–13 (2012)
61. Ishihara, H.: Reverse mathematics in Bishop's constructive mathematics. Philosophia Scientiae, Cahier special **6**, 43–59 (2006)
62. Jayne, J.E., Rogers, C.A.: First level Borel functions and isomorphisms. Journal de Mathématiques Pures et Appliquées. Neuvième Série **61**(2), 177–205 (1982)
63. Jockusch, C.G., Jr., Lewis, A.E.M.: Diagonally non-computable functions and bi-immunity. Journal of Symbolic Logic **78**(3), 977–988 (2013)
64. Jockusch, C.G., Jr., Soare, R.I.: Π^0_1 classes and degrees of theories. Transactions of the American Mathematical Society **173**, 33–56 (1972)
65. Kawamura, A.: Lipschitz continuous ordinary differential equations are polynomial-space complete. Computational Complexity **19**(2), 305–332 (2010)

66. Kawamura, A., Cook, S.: Complexity theory for operators in analysis. In: Proceedings of the 42nd ACM Symposium on Theory of Computing, *STOC '10*, pp. 495–502. ACM, New York (2010)
67. Kihara, T.: Decomposing Borel functions using the Shore-Slaman join theorem. Fundamenta Mathematicae **230**, 1–13 (2015)
68. Kihara, T.: All-or-unique choice and convex choice (2016), unpublished notes
69. Kihara, T., Pauly, A.: Dividing by zero - how bad is it, really? In: P. Faliszewski, A. Muscholl, R. Niedermeier (eds.) 41st International Symposium on Mathematical Foundations of Computer Science (MFCS 2016), *Leibniz International Proceedings in Informatics (LIPIcs)*, vol. 58, pp. 58:1–58:14. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2016)
70. Kleene, S.C.: Recursive functions and intuitionistic mathematics. In: Proceedings of the International Congress of Mathematicians, Cambridge, Mass., 1950, vol. 1, pp. 679–685. Amer. Math. Soc., Providence, R. I. (1952)
71. Kolmogoroff, A.: Zur Deutung der intuitionistischen Logik. Mathematische Zeitschrift **35**(1), 58–65 (1932)
72. Kreitz, C., Weihrauch, K.: Theory of representations. Theoretical Computer Science **38**, 35–53 (1985)
73. Kreitz, C., Weihrauch, K.: Compactness in constructive analysis revisited. Annals of Pure and Applied Logic **36**, 29–38 (1987)
74. Kudinov, O.V., Selivanov, V.L., Zhukov, A.V.: Undecidability in Weihrauch degrees. In: F. Ferreira, B. Löwe, E. Mayordomo, L. Mendes Gomes (eds.) Programs, Proofs, Processes, *Lecture Notes in Computer Science*, vol. 6158, pp. 256–265. Springer, Berlin (2010), 6th Conference on Computability in Europe, CiE 2010, Ponta Delgada, Azores, Portugal, June/July 2010
75. Kuyper, R.: On Weihrauch reducibility and intuitionistic reverse mathematics. Journal of Symbolic Logic **82**(4), 1438–1458 (2017)
76. Lacombe, D.: Classes récursivement fermés et fonctions majorantes. Comptes Rendus Académie des Sciences Paris **240**, 716–718 (1955)
77. Lacombe, D.: Les ensembles récursivement ouverts ou fermés, et leurs applications à l'Analyse récursive. Comptes Rendus Académie des Sciences Paris **245**, 1040–1043 (1957)
78. Lacombe, D.: Quelques procédés de définition en topologie récursive. In: A. Heyting (ed.) Constructivity in Mathematics, pp. 129–158. North-Holland, Amsterdam (1959), colloquium at Amsterdam, 1957
79. Le Roux, S., Pauly, A.: Finite choice, convex choice and finding roots. Logical Methods in Computer Science **11**(4), 4:6, 31 (2015)
80. Le Roux, S., Pauly, A.: Weihrauch degrees of finding equilibria in sequential games (extended abstract). In: A. Beckmann, V. Mitrana, M. Soskova (eds.) Evolving Computability, *Lecture Notes in Computer Science*, vol. 9136, pp. 246–257. Springer, Cham (2015), 11th Conference on Computability in Europe, CiE 2015, Bucharest, Romania, June 29–July 3, 2015
81. Moschovakis, Y.N.: Classical descriptive set theory as a refinement of effective descriptive set theory. Annals of Pure and Applied Logic **162**(3), 243–255 (2010)
82. Motto Ros, L.: Reduction games and reducibilities for sets of reals (abstract). In: U. Berger, V. Brattka, V. Selivanov, D. Spreen, H. Tsuiki (eds.) Computing with Infinite Data: Topological and Logical Foundations (Dagstuhl Seminar 11411), p. 28 (2012)
83. Mylatz, U.: Vergleich unstetiger Funktionen in der Analysis. Diplomarbeit. Fachbereich Informatik, FernUniversität Hagen (1992)
84. Mylatz, U.: Vergleich unstetiger Funktionen: "Principle of Omniscience" und Vollständigkeit in der *C*–hierarchie. Ph.D. thesis. Faculty for Mathematics and Computer Science, University Hagen, Hagen, Germany (2006)
85. Neumann, E.: Computational problems in metric fixed point theory and their Weihrauch degrees. Logical Methods in Computer Science **11**, 4:20,44 (2015)
86. Neumann, E., Pauly, A.: A topological view on algebraic computation models. Journal of Complexity **44**(Supplement C), 1–22 (2018)

87. Nobrega, H., Pauly, A.: Game characterizations and lower cones in the Weihrauch degrees. Logical Methods in Computer Science **15**(3), Paper No. 11, 29 (2019)
88. Orevkov, V.P.: A constructive mapping of the square onto itself displacing every constructive point (Russian). Doklady Akademii Nauk **152**, 55–58 (1963)
89. Patey, L.: The weakness of being cohesive, thin or free in reverse mathematics. Israel Journal of Mathematics **216**, 905–955 (2016)
90. Pauly, A.: Methoden zum Vergleich der Unstetigkeit von Funktionen. M.Sc. thesis. FernUniversität Hagen (2007)
91. Pauly, A.: How incomputable is finding Nash equilibria? Journal of Universal Computer Science **16**(18), 2686–2710 (2010)
92. Pauly, A.: On the (semi)lattices induced by continuous reducibilities. Mathematical Logic Quarterly **56**(5), 488–502 (2010)
93. Pauly, A.: Computable metamathematics and its application to game theory. Ph.D. thesis. University of Cambridge, Computer Laboratory, Clare College, Cambridge (2011)
94. Pauly, A.: The descriptive theory of represented spaces. arXiv 1408.5329 (2014)
95. Pauly, A.: On the topological aspects of the theory of represented spaces. Computability **5**(2), 159–180 (2016)
96. Pauly, A.: Many-one reductions and the category of multivalued functions. Mathematical Structures in Computer Science **27**(3), 376–404 (2017)
97. Pauly, A., de Brecht, M.: Non-deterministic computation and the Jayne-Rogers theorem. In: B. Löwe, G. Winskel (eds.) Proceedings 8th International Workshop on Developments in Computational Models, DCM 2012, Cambridge, United Kingdom, 17 June 2012., *Electronic Proceedings in Theoretical Computer Science*, vol. 143, pp. 87–96 (2014)
98. Pauly, A., de Brecht, M.: Descriptive set theory in the category of represented spaces. In: 30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), pp. 438–449 (2015)
99. Pauly, A., Fouché, W.L.: How constructive is constructing measures? Journal of Logic & Analysis **9**(c3), 1–44 (2017)
100. Pauly, A., Steinberg, F.: Comparing representations for function spaces in computable analysis. Theory of Computing Systems **62**, 557–582 (2018)
101. Pour-El, M.B., Richards, J.I.: Computability in Analysis and Physics. *Perspectives in Mathematical Logic*. Springer, Berlin (1989)
102. Rakotoniaina, T.: On the computational strength of Ramsey's theorem. Ph.D. thesis. Department of Mathematics and Applied Mathematics, University of Cape Town, Rondebosch, South Africa (2015)
103. Schröder, M.: Extended admissibility. Theoretical Computer Science **284**(2), 519–538 (2002)
104. Selivanov, V.L.: Difference hierarchy in φ–spaces. Algebra and Logic **43**(4), 238–248 (2004)
105. Semmes, B.T.: A game for the Borel functions. Ph.D. thesis. Institute for Logic, Language and Computation, University of Amsterdam (2009)
106. Simpson, S.G.: Subsystems of Second Order Arithmetic, 2nd edn. *Perspectives in Logic*. Cambridge University Press (2009)
107. Simpson, S.G.: Degrees of unsolvability: a tutorial. In: Evolving Computability, *Lecture Notes in Computer Science*, vol. 9136, pp. 83–94. Springer, Cham (2015)
108. Solomon, R.: Computable reductions and reverse mathematics. In: A. Beckmann, L. Bienvenu, N. Jonoska (eds.) Pursuit of the Universal, *Lecture Notes in Computer Science*, vol. 9709, pp. 182–191. Springer, Switzerland (2016), 12th Conference on Computability in Europe, CiE 2016, Paris, France, June 27 - July 1, 2016
109. Specker, E.: Der Satz vom Maximum in der rekursiven Analysis. In: A. Heyting (ed.) Constructivity in Mathematics, *Studies in Logic and the Foundations of Mathematics*, pp. 254–265. North-Holland, Amsterdam (1959), proc. Colloq., Amsterdam, Aug. 26–31, 1957
110. von Stein, T.: Vergleich nicht konstruktiv lösbarer Probleme in der Analysis. Diplomarbeit. Fachbereich Informatik, FernUniversität Hagen (1989)
111. Tavana, N.R., Weihrauch, K.: Turing machines on represented sets, a model of computation for analysis. Logical Methods in Computer Science **7**(2), 2:19, 21 (2011)

112. Uftring, P.: The characterization of Weihrauch reducibility in systems containing E-PA$^{\omega}$ + QF-AC^{0,0}. The Journal of Symbolic Logic (to appear) (2021)
113. Weihrauch, K.: Computability, *EATCS Monographs on Theoretical Computer Science*, vol. 9. Springer, Berlin (1987)
114. Weihrauch, K.: The degrees of discontinuity of some translators between representations of the real numbers. Technical Report TR-92-050, International Computer Science Institute, Berkeley (1992)
115. Weihrauch, K.: The TTE-interpretation of three hierarchies of omniscience principles. Informatik Berichte 130, FernUniversität Hagen, Hagen (1992)
116. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)
117. Weihrauch, K., Kreitz, C.: Representations of the real numbers and of the open subsets of the set of real numbers. Annals of Pure and Applied Logic **35**, 247–260 (1987)
118. Ziegler, M.: Real hypercomputation and continuity. Theory of Computing Systems **41**(1), 177–206 (2007)
119. Ziegler, M.: Revising type-2 computation and degrees of discontinuity. In: D. Cenzer, R. Dillhage, T. Grubba, K. Weihrauch (eds.) Proceedings of the Third International Conference on Computability and Complexity in Analysis, *Electronic Notes in Theoretical Computer Science*, vol. 167, pp. 255–274. Elsevier, Amsterdam (2007), CCA 2006, Gainesville, Florida, USA, November 1–5, 2006

There is an electronic version of a bibliography on Weihrauch complexity[8].

[8] See http://cca-net.de/publications/weibib.php

Index

V. Brattka, P. Hertling (eds.), *Handbook of Computability and Complexity in Analysis*,
Theory and Applications of Computability, https://doi.org/10.1007/978-3-030-59234-9

Q

R